



# jQuery *mobile*

**tutorialspoint**  
SIMPLY EASY LEARNING

[www.tutorialspoint.com](http://www.tutorialspoint.com)



<https://www.facebook.com/tutorialspointindia>



<https://twitter.com/tutorialspoint>

## About the Tutorial

---

JQuery Mobile is a user interface framework, built on jQuery Core and used for developing responsive websites or applications that are accessible on mobile, tablet, and desktop devices. It uses features of both jQuery and jQueryUI to provide API features for mobile web applications.

This tutorial will teach you the basics of jQuery Mobile framework. We will also discuss some detailed concepts related to jQuery Mobile.

## Audience

---

This tutorial has been prepared for beginners to help them understand basic jQuery Mobile framework. After completing this tutorial, you will find yourself at a moderate level of expertise in jQuery Mobile framework from where you can take yourself to the next levels.

## Prerequisites

---

jQuery Mobile framework is based on JavaScript programming language. Hence, if you have a basic understanding on JavaScript programming, then it will be fun to learn jQuery Mobile framework.

## Copyright & Disclaimer

---

© Copyright 2017 by Tutorials Point (I) Pvt. Ltd.

All the content and graphics published in this e-book are the property of Tutorials Point (I) Pvt. Ltd. The user of this e-book is prohibited to reuse, retain, copy, distribute or republish any contents or a part of contents of this e-book in any manner without written consent of the publisher.

We strive to update the contents of our website and tutorials as timely and as precisely as possible, however, the contents may contain inaccuracies or errors. Tutorials Point (I) Pvt. Ltd. provides no guarantee regarding the accuracy, timeliness or completeness of our website or its contents including this tutorial. If you discover any errors on our website or in this tutorial, please notify us at [contact@tutorialspoint.com](mailto:contact@tutorialspoint.com)

## Table of Contents

---

|   |    |
|---|----|
| <b>About the Tutorial.....</b>                | i  |
| <b>Audience .....</b>                         | i  |
| <b>Prerequisites .....</b>                    | i  |
| <b>Copyright &amp; Disclaimer.....</b>        | i  |
| <b>Table of Contents .....</b>                | ii |
| <br>1. JQUERY MOBILE – OVERVIEW .....         | 1  |
| <b>Why Use jQuery Mobile? .....</b>           | 1  |
| <b>Features of jQuery Mobile .....</b>        | 1  |
| <b>Advantages of jQuery Mobile.....</b>       | 1  |
| <b>Disadvantages of jQuery Mobile .....</b>   | 2  |
| <br>2. JQUERY MOBILE – SETUP.....             | 3  |
| <b>Download jQuery Mobile .....</b>           | 3  |
| <b>Download jQuery Library from CDNs.....</b> | 10 |
| <br>3. JQUERY MOBILE – PAGES .....            | 13 |
| <b>Single Page .....</b>                      | 13 |
| <b>Multi-Page Template.....</b>               | 16 |
| <b>Dialogs Page.....</b>                      | 18 |
| <b>Basic Dialogs .....</b>                    | 18 |
| <b>Transitions Dialogs .....</b>              | 20 |
| <b>Closing Dialogs .....</b>                  | 23 |
| <b>Chaining Dialogs.....</b>                  | 25 |
| <b>Styling &amp; Theming Dialogs.....</b>     | 28 |
| <b>Dialog Width and Margins.....</b>          | 31 |
| <b>Prefetching Pages.....</b>                 | 32 |

|                                     |    |
|-------------------------------------|----|
| 4. JQUERY MOBILE – ICONS.....       | 35 |
| Icon Set .....                      | 35 |
| Icon Positioning.....               | 37 |
| Icon Only.....                      | 38 |
| Icon Shadow.....                    | 40 |
| Removing Icon Circle .....          | 41 |
| Black and White Icon Sets .....     | 42 |
| Combining alt and nodisc .....      | 44 |
| 5. JQUERY MOBILE – TRANSITIONS..... | 46 |
| Fade Page Transition .....          | 47 |
| Fade Dialog Transition.....         | 49 |
| Flip Page Transition .....          | 51 |
| Flip Dialog Transition.....         | 53 |
| Page Popup Transition .....         | 54 |
| Dialog Popup Transition .....       | 56 |
| Page Flow Transition .....          | 58 |
| Dialog Flow Transition.....         | 60 |
| Page Slide Transition .....         | 62 |
| Dialog Slide Transition.....        | 64 |
| Page Slidedefade Transition .....   | 66 |
| Dialog Slidedefade Transition.....  | 68 |
| Page Slideup Transition .....       | 70 |
| Dialog Slideup Transition.....      | 72 |
| Page Slidedown Transition .....     | 74 |
| Dialog Slidedown Transition.....    | 76 |
| Page Turn Transition .....          | 78 |

|   |            |
|---|------------|
| <b>Dialog Turn Transition .....</b>                     | <b>80</b>  |
| <b>Page None Transition .....</b>                       | <b>82</b>  |
| <b>Dialog None Transition.....</b>                      | <b>84</b>  |
| <b>6. JQUERY MOBILE – LAYOUTS .....</b>                 | <b>88</b>  |
| <b>Grid Layouts .....</b>                               | <b>88</b>  |
| <b>Basic Grid Layouts .....</b>                         | <b>89</b>  |
| <b>Two-column Grids .....</b>                           | <b>90</b>  |
| <b>Three-column Grids.....</b>                          | <b>92</b>  |
| <b>Four-column Grids.....</b>                           | <b>93</b>  |
| <b>Five-column Grids .....</b>                          | <b>95</b>  |
| <b>Multiple Row Grid .....</b>                          | <b>97</b>  |
| <b>Grid Solo Class.....</b>                             | <b>99</b>  |
| <b>Responsive Grids .....</b>                           | <b>100</b> |
| <b>Breakpoint Preset .....</b>                          | <b>101</b> |
| <b>Buttons in Grids .....</b>                           | <b>103</b> |
| <b>Basic Buttons in Grids.....</b>                      | <b>104</b> |
| <b>Mini-sized Buttons in Grids .....</b>                | <b>106</b> |
| <b>Icon Only, Inline Centered Buttons in Grids.....</b> | <b>108</b> |
| <b>Responsive Buttons in Grids.....</b>                 | <b>109</b> |
| <b>Alignment Buttons in Grids .....</b>                 | <b>111</b> |
| <b>Custom Responsive Grid .....</b>                     | <b>112</b> |
| <b>7. JQUERY MOBILE – WIDGETS.....</b>                  | <b>116</b> |
| <b>Buttons .....</b>                                    | <b>118</b> |
| <b>Basic Buttons .....</b>                              | <b>118</b> |
| <b>Rounded Buttons .....</b>                            | <b>120</b> |

|  |            |
|--|------------|
| <b>Grouped Buttons.....</b>                | <b>121</b> |
| <b>Shadow Buttons.....</b>                 | <b>123</b> |
| <b>Inline Buttons.....</b>                 | <b>124</b> |
| <b>Mini Button.....</b>                    | <b>127</b> |
| <b>Back Button.....</b>                    | <b>128</b> |
| <b>Icon Position .....</b>                 | <b>130</b> |
| <b>Disabled Button .....</b>               | <b>131</b> |
| <b>Checkbox Widget .....</b>               | <b>133</b> |
| <b>Checkbox Basic markup.....</b>          | <b>134</b> |
| <b>Checkbox Mini Size .....</b>            | <b>135</b> |
| <b>Checkbox Vertical Group.....</b>        | <b>136</b> |
| <b>Checkbox Horizontal Group.....</b>      | <b>138</b> |
| <b>Checkbox Icon Position .....</b>        | <b>139</b> |
| <b>Checkbox Theme.....</b>                 | <b>140</b> |
| <b>Checkbox Disabled .....</b>             | <b>142</b> |
| <b>Checkbox Enhanced .....</b>             | <b>143</b> |
| <b>Radio Widget .....</b>                  | <b>144</b> |
| <b>Basic Radio Widget.....</b>             | <b>145</b> |
| <b>Basic Radio Mini Size Widget.....</b>   | <b>147</b> |
| <b>Radio Vertical Group Widget.....</b>    | <b>148</b> |
| <b>Radio Horizontal Group Widget .....</b> | <b>149</b> |
| <b>Radio Icon Position Widget .....</b>    | <b>151</b> |
| <b>Radio Widget Theme .....</b>            | <b>152</b> |
| <b>Radio Disabled Widget .....</b>         | <b>154</b> |
| <b>Radio Enhanced Widget .....</b>         | <b>155</b> |
| <b>Datapicker Widget .....</b>             | <b>156</b> |

|  |            |
|--|------------|
| <b>Collapsible Widget .....</b>              | <b>159</b> |
| <b>Collapsible.....</b>                      | <b>160</b> |
| <b>Basic Collapsible.....</b>                | <b>161</b> |
| <b>Collapsible Theme .....</b>               | <b>162</b> |
| <b>Collapsible Expanded .....</b>            | <b>163</b> |
| <b>Mini Sized Collapsible.....</b>           | <b>165</b> |
| <b>Collapsible Icons.....</b>                | <b>166</b> |
| <b>Collapsible Icons Positioning .....</b>   | <b>167</b> |
| <b>Collapsible Legend .....</b>              | <b>168</b> |
| <b>Non-inset Collapsible .....</b>           | <b>170</b> |
| <b>Collapsible Set.....</b>                  | <b>171</b> |
| <b>Collapsible Pre-rendered Markup.....</b>  | <b>173</b> |
| <b>Dynamic Collapsible .....</b>             | <b>174</b> |
| <b>Collapsible Set.....</b>                  | <b>176</b> |
| <b>Collapsible Markup .....</b>              | <b>177</b> |
| <b>Collapsible Inset vs. Full Width.....</b> | <b>179</b> |
| <b>jQuery Mobile - Listview .....</b>        | <b>179</b> |
| <b>Listview Unordered Widget.....</b>        | <b>180</b> |
| <b>Listview Ordered .....</b>                | <b>182</b> |
| <b>Listview Linked.....</b>                  | <b>183</b> |
| <b>Listview Inset .....</b>                  | <b>184</b> |
| <b>Listview Filter .....</b>                 | <b>185</b> |
| <b>Listview Filter Reveal .....</b>          | <b>187</b> |
| <b>Listview Dividers .....</b>               | <b>188</b> |
| <b>Listview Autodividers .....</b>           | <b>190</b> |
| <b>Listview Count Bubbles .....</b>          | <b>191</b> |

|   |            |
|---|------------|
| <b>Listview Standard Icons.....</b>                 | <b>192</b> |
| <b>Listview Icons: 16x16.....</b>                   | <b>194</b> |
| <b>Listview Thumbnails.....</b>                     | <b>196</b> |
| <b>Listview Split buttons .....</b>                 | <b>198</b> |
| <b>Listview Formatted Content.....</b>              | <b>200</b> |
| <b>Listview Theme .....</b>                         | <b>203</b> |
| <b>Listview Forms .....</b>                         | <b>204</b> |
| <b>Listview Collapsible .....</b>                   | <b>207</b> |
| <b>Grouped Collapsible With Listviews .....</b>     | <b>208</b> |
| <b>Full Width Collapsible Listview .....</b>        | <b>213</b> |
| <b>Mini Collapsible Set.....</b>                    | <b>219</b> |
| <b>Collapsible Set Icon .....</b>                   | <b>221</b> |
| <b>Collapsible Set Icon Positioning .....</b>       | <b>223</b> |
| <b>Collapsible Set Without Rounded Corners.....</b> | <b>224</b> |
| <b>Collapsible Set Theme .....</b>                  | <b>226</b> |
| <b>Nested Collapsible.....</b>                      | <b>228</b> |
| <b>jQuery Mobile – Controlgroup Widget .....</b>    | <b>229</b> |
| <b>Controlgroup Vertical Widget.....</b>            | <b>230</b> |
| <b>Controlgroup Horizontal Widget .....</b>         | <b>231</b> |
| <b>Controlgroup Minimized Widget.....</b>           | <b>233</b> |
| <b>Controlgroup Radio Button .....</b>              | <b>234</b> |
| <b>Controlgroup Checkbox.....</b>                   | <b>237</b> |
| <b>Controlgroup Selects .....</b>                   | <b>240</b> |
| <b>Controlgroup Textinputs .....</b>                | <b>243</b> |
| <b>Controlgroup Pre-Rendered Markup .....</b>       | <b>244</b> |
| <b>Fliterable Widget.....</b>                       | <b>245</b> |

|  |            |
|--|------------|
| <b>Filterable Basics .....</b>                                 | <b>246</b> |
| <b>Table Filter .....</b>                                      | <b>248</b> |
| <b>Controlgroup Filter.....</b>                                | <b>251</b> |
| <b>Filter Collapsible Set.....</b>                             | <b>252</b> |
| <b>Filter Collapsible Set &amp; Collapsible Children .....</b> | <b>254</b> |
| <b>Filter Anything .....</b>                                   | <b>257</b> |
| <b>Filter Styling .....</b>                                    | <b>259</b> |
| <b>Filter Reveal .....</b>                                     | <b>260</b> |
| <b>Filter Custom Callback.....</b>                             | <b>262</b> |
| <b>Filterable Pre-rendering .....</b>                          | <b>263</b> |
| <b>Flip Switch Widget .....</b>                                | <b>264</b> |
| <b>Listview Widget.....</b>                                    | <b>269</b> |
| <b>Listview .....</b>  | <b>270</b> |
| <b>Listview Unordered Widget.....</b>                          | <b>272</b> |
| <b>Listview Ordered .....</b>                                  | <b>273</b> |
| <b>Listview Linked.....</b>                                    | <b>274</b> |
| <b>Listview Inset .....</b>                                    | <b>275</b> |
| <b>Listview Filter .....</b>                                   | <b>277</b> |
| <b>Listview Filter Reveal .....</b>                            | <b>278</b> |
| <b>Listview Dividers .....</b>                                 | <b>280</b> |
| <b>Listview Autodividers .....</b>                             | <b>281</b> |
| <b>Listview Count bubbles .....</b>                            | <b>282</b> |
| <b>Listview Standard Icons.....</b>                            | <b>283</b> |
| <b>Listview Icons: 16x16.....</b>                              | <b>285</b> |
| <b>Listview Thumbnails.....</b>                                | <b>287</b> |
| <b>Listview Split Buttons.....</b>                             | <b>289</b> |

|   |            |
|---|------------|
| <b>Listview Formatted Content .....</b>               | <b>292</b> |
| <b>Listview Theme .....</b>                           | <b>294</b> |
| <b>Listview Forms .....</b>                           | <b>296</b> |
| <b>Listview Collapsible .....</b>                     | <b>298</b> |
| <b>Grouped Collapsible With Listviews .....</b>       | <b>300</b> |
| <b>Full Width Collapsible Listview .....</b>          | <b>305</b> |
| <b>Listview Autocomplete.....</b>                     | <b>309</b> |
| <b>Listview Remote Autocomplete .....</b>             | <b>310</b> |
| <b>Listview Local Data.....</b>                       | <b>312</b> |
| <b>Listview Providing Richer Search Content .....</b> | <b>315</b> |
| <b>Listview Remote Autocomplete .....</b>             | <b>316</b> |
| <b>Listview Autodivider Linkbar .....</b>             | <b>319</b> |
| <b>Listview Autodivider Selector.....</b>             | <b>322</b> |
| <b>Nested Listviews .....</b>                         | <b>324</b> |
| <b>Listview Collapsible Item Flat .....</b>           | <b>328</b> |
| <b>Listview Collapsible Item Indented.....</b>        | <b>332</b> |
| <b>Listview Responsive Grid.....</b>                  | <b>336</b> |
| <b>Loader Widget.....</b>                             | <b>342</b> |
| <b>Standard Loader Widget .....</b>                   | <b>342</b> |
| <b>Custom HTML Widget .....</b>                       | <b>344</b> |
| <b>Theme Widget.....</b>                              | <b>346</b> |
| <b>Navbar Widget .....</b>                            | <b>347</b> |
| <b>Navbar Basics .....</b>                            | <b>349</b> |
| <b>Navbar Multi-row.....</b>                          | <b>351</b> |
| <b>Navbars in Headers .....</b>                       | <b>353</b> |
| <b>Navbars in Footers .....</b>                       | <b>354</b> |

|  |            |
|--|------------|
| <b>Navbars Persistent .....</b>          | <b>355</b> |
| <b>Navbars Icons.....</b>                | <b>358</b> |
| <b>Navbars Icons Position .....</b>      | <b>360</b> |
| <b>Navbars 3rd Party Icon Sets.....</b>  | <b>362</b> |
| <b>Navbars Themes.....</b>               | <b>364</b> |
| <b>Navbars with Button Elements.....</b> | <b>366</b> |
| <b>Panels Widget .....</b>               | <b>368</b> |
| <b>Panels .....</b>                      | <b>368</b> |
| <b>Closing Panel.....</b>                | <b>372</b> |
| <b>External Panel .....</b>              | <b>373</b> |
| <b>Fixed Panel Positioning .....</b>     | <b>376</b> |
| <b>Panel Responsive .....</b>            | <b>378</b> |
| <b>Panel Styling.....</b>                | <b>381</b> |
| <b>Panel Swipe.....</b>                  | <b>383</b> |
| <b>Internal and External Panels.....</b> | <b>385</b> |
| <b>Popup Widget .....</b>                | <b>387</b> |
| <b>Basic Popup.....</b>                  | <b>388</b> |
| <b>Basics of Popup .....</b>             | <b>389</b> |
| <b>Popup Tooltip.....</b>                | <b>390</b> |
| <b>Closing Popups .....</b>              | <b>392</b> |
| <b>Photo Lightbox.....</b>               | <b>394</b> |
| <b>Popup Menu .....</b>                  | <b>397</b> |
| <b>Popup Form .....</b>                  | <b>398</b> |
| <b>Popup Dialog .....</b>                | <b>400</b> |
| <b>Popup Padding .....</b>               | <b>402</b> |
| <b>Popup Position .....</b>              | <b>403</b> |

|  |            |
|--|------------|
| <b>Popup Transitions .....</b>             | <b>405</b> |
| <b>Popup Theme .....</b>                   | <b>407</b> |
| <b>Popup Arrow .....</b>                   | <b>409</b> |
| <b>Pre-rendered Popup Markup .....</b>     | <b>411</b> |
| <b>Popup Alignment .....</b>               | <b>412</b> |
| <b>Dynamic Popup .....</b>                 | <b>414</b> |
| <b>iframes in Popup .....</b>              | <b>417</b> |
| <b>iframe Video .....</b>                  | <b>418</b> |
| <b>Map Popup .....</b>                     | <b>420</b> |
| <b>Scaling Images .....</b>                | <b>423</b> |
| <b>Popup on Multiple Pages .....</b>       | <b>426</b> |
| <b>Range Slider Widget .....</b>           | <b>429</b> |
| <b>Basic Rangeslider .....</b>             | <b>430</b> |
| <b>Rangeslider Step Attribute .....</b>    | <b>431</b> |
| <b>Rangeslider No Highlight .....</b>      | <b>433</b> |
| <b>Rangeslider Theme .....</b>             | <b>434</b> |
| <b>Rangeslider Mini .....</b>              | <b>435</b> |
| <b>Rangeslider Disabled .....</b>          | <b>436</b> |
| <b>Rangeslider Label Hidden .....</b>      | <b>438</b> |
| <b>Rangeslider Fieldcontain .....</b>      | <b>439</b> |
| <b>Rangeslider Fieldcontain Mini .....</b> | <b>440</b> |
| <b>Table Widget .....</b>                  | <b>442</b> |
| <b>Column Toggle .....</b>                 | <b>443</b> |
| <b>Column Toggle Heading Groups .....</b>  | <b>447</b> |
| <b>Column Toggle Customization .....</b>   | <b>451</b> |
| <b>Table Reflow .....</b>                  | <b>455</b> |

|  |            |
|--|------------|
| <b>Table Reflow Heading Groups .....</b>       | <b>458</b> |
| <b>Table Reflow Stripes and Strokes .....</b>  | <b>461</b> |
| <b>Table Reflow Custom Styles .....</b>        | <b>465</b> |
| <b>Tabs Widget .....</b>                       | <b>469</b> |
| <b>Use Navbar for Tabs .....</b>               | <b>470</b> |
| <b>Tabbed Page Content .....</b>               | <b>472</b> |
| <b>Textinput Widget .....</b>                  | <b>474</b> |
| <b>Textinput Text and Textarea .....</b>       | <b>476</b> |
| <b>Textinput Search .....</b>                  | <b>477</b> |
| <b>Textinput Number .....</b>                  | <b>478</b> |
| <b>Textinput Date, Month, and Week .....</b>   | <b>480</b> |
| <b>Textinput Time and Color .....</b>          | <b>481</b> |
| <b>Textinput Telephone and Email .....</b>     | <b>483</b> |
| <b>Textinput URL and Password .....</b>        | <b>484</b> |
| <b>Textinput Datetime .....</b>                | <b>486</b> |
| <b>Textinput File .....</b>                    | <b>487</b> |
| <b>Textinput Mini .....</b>                    | <b>488</b> |
| <b>Textinput Placeholder .....</b>             | <b>490</b> |
| <b>Textinput Value/Text .....</b>              | <b>491</b> |
| <b>Textinput Label Hidden .....</b>            | <b>493</b> |
| <b>Textinput Disabled .....</b>                | <b>494</b> |
| <b>Textinput Enhanced .....</b>                | <b>496</b> |
| <b>Textinput Fieldcontain .....</b>            | <b>497</b> |
| <b>Textinput Fieldcontain, Mini Size .....</b> | <b>499</b> |
| <b>Toolbar Widget .....</b>                    | <b>501</b> |
| <b>Toolbar .....</b>                           | <b>502</b> |

|   |            |
|---|------------|
| <b>Toolbar Markup .....</b>               | <b>502</b> |
| <b>Toolbar Theme .....</b>                | <b>504</b> |
| <b>Buttons in Toolbars .....</b>          | <b>505</b> |
| <b>Buttons in Footers.....</b>            | <b>507</b> |
| <b>Adding Back Button to Header .....</b> | <b>508</b> |
| <b>Navbar Widget .....</b>                | <b>510</b> |
| <b>Navbar Basics .....</b>                | <b>511</b> |
| <b>Navbar Multi-row.....</b>              | <b>514</b> |
| <b>Navbars in Headers .....</b>           | <b>516</b> |
| <b>Navbars in Footers .....</b>           | <b>517</b> |
| <b>Navbars Persistent .....</b>           | <b>518</b> |
| <b>Navbars Icons.....</b>                 | <b>522</b> |
| <b>Navbars Icons Position .....</b>       | <b>523</b> |
| <b>Navbars 3rd Party Icon .....</b>       | <b>525</b> |
| <b>Navbars Themes.....</b>                | <b>527</b> |
| <b>Navbars with Button Elements.....</b>  | <b>529</b> |
| <b>Grouped Buttons.....</b>               | <b>530</b> |
| <b>Adding a Padding .....</b>             | <b>532</b> |
| <b>Dynamic Toolbars.....</b>              | <b>533</b> |
| <b>External Toolbars .....</b>            | <b>535</b> |
| <b>Fixed Position Toolbars .....</b>      | <b>537</b> |
| <b>Fullscreen Position Toolbars.....</b>  | <b>541</b> |
| <b>External Fixed Toolbars .....</b>      | <b>545</b> |
| <b>Persistent Toolbars .....</b>          | <b>549</b> |
| <b>Navbars Persistent .....</b>           | <b>549</b> |
| <b>Ajax Optimized Toolbars .....</b>      | <b>553</b> |

|   |            |
|---|------------|
| <b>Form in Toolbar.....</b>             | <b>555</b> |
| <b>8. JQUERY MOBILE – EVENTS.....</b>   | <b>560</b> |
| <b>Mobile Events .....</b>              | <b>560</b> |
| <b>Touch Events .....</b>               | <b>562</b> |
| <b>Touch Tap Event.....</b>             | <b>563</b> |
| <b>Touch Taphold Event.....</b>         | <b>564</b> |
| <b>Touch Swipe Event .....</b>          | <b>566</b> |
| <b>Touch Swipeleft Event.....</b>       | <b>567</b> |
| <b>Touch Swiperight Event.....</b>      | <b>569</b> |
| <b>Scroll Events.....</b>               | <b>571</b> |
| <b>Scrollstart Event .....</b>          | <b>571</b> |
| <b>Scrollstop Event .....</b>           | <b>574</b> |
| <b>Orientation Event.....</b>           | <b>577</b> |
| <b>Page Events.....</b>                 | <b>579</b> |
| <b>Page Initialization Events .....</b> | <b>579</b> |
| <b>Page Load Events .....</b>           | <b>581</b> |
| <b>Page Transition Events .....</b>     | <b>583</b> |
| <b>9. JQUERY MOBILE – FORMS .....</b>   | <b>586</b> |
| <b>Form Basic.....</b>                  | <b>586</b> |
| <b>Form Structure.....</b>              | <b>587</b> |
| <b>Form Buttons .....</b>               | <b>589</b> |
| <b>jQuery Mobile – Icon Set .....</b>   | <b>590</b> |
| <b>Form Field Container.....</b>        | <b>593</b> |
| <b>Form Inputs.....</b>                 | <b>595</b> |
| <b>Form Select .....</b>                | <b>597</b> |

|   |            |
|---|------------|
| <b>Form Select Menus.....</b>                                   | <b>598</b> |
| <b>Form Custom Select Menus.....</b>                            | <b>599</b> |
| <b>Form Multiple Selection .....</b>                            | <b>601</b> |
| <b>Form Slider.....</b>   | <b>603</b> |
| <b>Form Slider Controls.....</b>                                | <b>603</b> |
| <b>Form Flip Toggle Switch .....</b>                            | <b>605</b> |
| <b>Refreshing and Auto Initialization of Form Elements.....</b> | <b>606</b> |
| <b>10. JQUERY MOBILE – THEMES .....</b>                         | <b>608</b> |
| <b>Themes Types .....</b>                                       | <b>609</b> |
| <b>Styling &amp; Theming Dialogs.....</b>                       | <b>611</b> |
| <b>Theming Buttons, Icons and Popups .....</b>                  | <b>614</b> |
| <b>Buttons in Footer .....</b>                                  | <b>616</b> |
| <b>Navbars Themes.....</b>                                      | <b>618</b> |
| <b>Theming Panels.....</b>                                      | <b>620</b> |
| <b>Theming Collapsible Button and Split Buttons.....</b>        | <b>622</b> |
| <b>Theming Lists and Collapsible Lists .....</b>                | <b>625</b> |
| <b>Collapsible Forms .....</b>                                  | <b>628</b> |
| <b>11. JQUERY MOBILE – CSS CLASSES.....</b>                     | <b>630</b> |
| <b>Global Classes .....</b>                                     | <b>630</b> |
| <b>Button Classes.....</b>                                      | <b>630</b> |
| <b>Icon Classes .....</b>                                       | <b>631</b> |
| <b>Theme Classes.....</b>                                       | <b>633</b> |
| <b>Grid Classes .....</b>                                       | <b>634</b> |

|  |     |
|--|-----|
| 12. JQUERY MOBILE – DATA ATTRIBUTES..... | 635 |
| Button.....                              | 635 |
| Checkbox.....                            | 636 |
| Collapsible.....                         | 636 |
| Collapsible Set.....                     | 637 |
| Controlgroup.....                        | 638 |
| Dialog.....                              | 638 |
| Enhancement .....                        | 639 |
| Fixed Toolbar .....                      | 639 |
| Flip Toggle Switch.....                  | 640 |
| Footer .....                             | 640 |
| Header .....                             | 641 |
| Inputs.....                              | 641 |
| Link .....                               | 642 |
| List .....                               | 642 |
| List Item .....                          | 643 |
| Navbar .....                             | 644 |
| Page .....                               | 644 |
| Popup.....                               | 644 |
| Radio Button .....                       | 645 |
| Select .....                             | 646 |
| Slider.....                              | 647 |

# 1. jQuery Mobile – Overview

JQuery Mobile is a user interface framework, which is built on jQuery Core and used for developing responsive websites or applications that are accessible on mobile, tablet, and desktop devices. It uses the features of both jQuery and jQuery UI to provide API features for mobile web applications.

It was developed by the jQuery project team in the year 2010 and written in JavaScript.

## Why Use jQuery Mobile?

---

- It creates web applications that it will work the same way on the mobile, tablet, and desktop devices.
- It is compatible with other frameworks such as PhoneGap, Whitelight, etc.
- It provides a set of touch-friendly form inputs and UI widgets.
- The progressive enhancement brings a unique functionality to all mobile, tablet, and desktop platforms and adds efficient page loads and wider device support.

## Features of jQuery Mobile

---

- It is built on jQuery Core and "write less, do more" UI framework.
- It is an open source framework, and cross-platform as well as cross-browser compatible.
- It is written in JavaScript and uses features of both jQuery and jQuery UI for building mobile-friendly sites.
- It integrates HTML5, CCS3, jQuery and jQuery UI into one framework for creating pages with minimal scripting.
- It includes Ajax navigation system that uses animated page transitions.

## Advantages of jQuery Mobile

---

- It is easy to learn and develop applications if you have knowledge of HTML5, CSS3 features.
- It is cross-platform and cross-browser compatible so you don't have to worry about writing different code for each device resolution.
- You can create the custom theme using ThemeRoller without writing the line of code. It supports all HTML5 browsers.

- It uses HTML5 along with JavaScript for easy development of web applications.
- It is built in a way that allows the same code to automatically scale from the mobile screen to desktop screen.

## Disadvantages of jQuery Mobile

---

- There are limited options for CSS themes, so sites can look similar which are built by these themes.
- Applications which are developed using jQuery Mobile are slower on mobiles.
- It becomes more time consuming when you combine jQuery mobile with other mobile frameworks.
- Difficult to provide complete customized visual design.
- All the features in a device cannot be accessed by JavaScript in a browser.

## 2. jQuery Mobile – Setup

In this chapter, we will discuss how to install and set up jQuery Mobile.

### Download jQuery Mobile

When you open the link [jquerymobile.com/](http://jquerymobile.com/), you will see there are two options to download jQuery mobile library.



- **Custom Download** - Click this button to download a customized version of library.
- **Latest Stable** - Click this button to get the stable and latest version of jQuery mobile library.

### Custom Download with Download Builder

Using Download Builder, you can create a custom build including only the portions of the library that you need. When you download this new customized version of jQuery Mobile, you will see the following screen.

The screenshot shows the jQuery Mobile Download Builder interface. At the top, there's a navigation bar with links for Demos, Download, API Documentation, Themes, Resources, Blog, and About. A search bar is also present. On the right side of the header, there's a green heart-shaped button with the text "Your donations help fund the continued development and growth of jQuery." and a "SUPPORT THE PROJECT" button.

## jQuery Mobile Download Builder

Customize your jQuery Mobile download by selecting the specific modules you need in the form below. For standard configurations, [download or link to CDN versions](#) of pre-built packages. Please note that the jQuery Mobile Download Builder is still in alpha, and as such should not be used on production websites.

Please report any issues you might find in the [issue tracker](#), and thanks!

**Select branch**

1.4.5 ▾

**Core** Select all

- Animation Complete  
*A handler for css transition & animation end events to ensure callback is executed.*
- jqmData  
*Mobile versions of Data functions to allow for namespacing.*
- Defaults  
*Default values for jQuery Mobile.*
- Helpers  
*Helper functions and references.*
- Init  
*Global initialization of the library.*
- Namespace  
*The mobile namespace on the jQuery object.*
- Support Tests  
*Assorted tests to qualify browsers by detecting features.*
- Orientation support test  
*Feature test for orientation.*

- Touch support test**  
*Touch feature test*
- Virtual Mouse (vmouse) Bindings**  
*Normalizes touch/mouse events.*
- Widget Factory**  
*Widget factory extensions for mobile.*
- Page Creation**  
*Basic page definition and formatting.*

**Events****Select all \***

- Navigate**  
*Provides a wrapper around hashchange and popstate*
- Orientation Change**  
*Provides a wrapper around the inconsistent browser implementations of orientationchange*
- Throttled Resize**  
*Fires a resize event with a slight delay to prevent excessive callback invocation*
- Touch**  
*Touch events including: touchstart, touchmove, touchend, tap, taphold, swipe, swipeleft, swiperight, scrollstart, scrollstop*
- Events**  
*Custom events and shortcuts.*

**Forms****Select all \***

- Buttons: Link-based**  
*Applies button styling to links*
- Fieldcontainers**  
*Styling to responsively position forms and labels based on screen width and add visual separation*
- Controlgroups**  
*Visually groups sets of buttons, checks, radios, etc.*
- Textarea Autosize**  
*Enhances and consistently styles text inputs.*

- Buttons: Input or button-based  
*Custom-styled native input/buttons*
- Checkboxes & Radio Buttons  
*Consistent styling for checkboxes/radio buttons.*
- Text Input Clear Button  
*Add the ability to have a clear button*
- Flip Switch  
*Consistent styling for native select menus. Tapping opens a native select menu.*
- Range Slider  
*Range Slider form widget*
- Form Reset
- Selects  
*Consistent styling for native select menus. Tapping opens a native select menu.*
- Selects: Custom menus  
*Extension to select menus to support menu styling, placeholder options, and multi-select features.*
- Slider  
*Slider form widget*
- Slidertooltip  
*Slider tooltip extension*
- Text Inputs & Textareas  
*Enhances and consistently styles text inputs.*

**Navigation**[Select all \\*](#)

- Content Management  
*Applies the AJAX navigation system to links and forms to enable page transitions*
- Base Tag  
*Dynamic Base Tag Support*

- History Manager  
*Manages a stack of history entries. Used exclusively by the Navigation Manager.*
- Navigate Method  
*A wrapper for the primary Navigator and History objects in jQuery Mobile*
- Navigation Manager  
*Manages URL history and information in conjunction with the navigate event*
- Path Helpers  
*Path parsing and manipulation helpers*
- Content Management  
*Widget to create page container which manages pages and transitions*

#### Transitions

Select all

- Transition Concurrent  
*Animated page change with concurrent transition style application*
- Transition Handlers  
*Animated page change handlers for integrating with Navigation*
- Transition Serial  
*Animated page change with serial transition style application*
- Transition Core  
*Animated page change base constructor and logic*
- All Transitions  
*All the stock transitions and associated CSS*
- Flip Transition  
*Animation styles and fallback transitions definition for non-3D supporting browsers*
- Flow Transition  
*Animation styles and fallback transitions definition for non-3D supporting browsers*

- Pop Transition**  
*Animation styles and fallback transitions definition for non-3D supporting browsers*
- Slide Transition**  
*Animation styles and fallback transitions definition for non-3D supporting browsers*
- Slidedown Transition**  
*Animation styles and fallback transitions definition for non-3D supporting browsers*
- Slidefade Transition**  
*Animation styles and fallback transitions definition for non-3D supporting browsers*
- Slideup Transition**  
*Animation styles and fallback transitions definition for non-3D supporting browsers*
- Turn Transition**  
*Animation styles and fallback transitions definition for non-3D supporting browsers*

**Utilities**[Select all !\[\]\(90c859a17dbc6c3879e6b0c04b61632c\_img.jpg\)](#)

- Degrade Inputs**  
*Changes input type to another after custom enhancements are made (ex: range > numeric)*
- Link Classes**  
*Adds classes to links.*
- Match Media Polyfill**  
*A workaround for browsers without window.matchMedia*
- "nojs" Classes**  
*Adds class to make elements hidden to A grade browsers*
- Zoom Handling**  
*Utility methods for enabling and disabling user scaling (pinch zoom)*
- iOS Orientation Change Fix**  
*Fixes the orientation change bug in iOS when switching between landscape and portrait*

**Widgets**

Select all

- Grid Layouts (Columns)**  
Applies classes for creating grid or column styling
- First & Last Classes**  
Behavior mixin to mark first and last visible item with special classes.
- Collapsible**  
Creates collapsible content blocks.
- Collapsible Sets (Accordions)**  
For creating grouped collapsible content areas.
- Dialogs (deprecated)**  
Displays a page as a modal dialog with inset appearance and overlay background
- Filterable**  
Renders the children of an element filterable via a callback and a `lndput`
- Filterable-widgetlink**  
Links options present in the widget to be filtered to the input.
- Table: Column Toggle**  
Extends the table widget to a column toggle menu and responsive column visibility
- Table: reflow**  
Extends the table widget to reflow on narrower screens
- Tabs**  
jQuery UI Tabs widget now for mobile!
- Toolbars: Fixed**  
Behavior for "fixed" headers and footers - be sure to also include the item 'Browser specific workarounds for "fixed" headers and footers' when supporting Android 2.x or iOS 5

**Build My Download**

Notes: The zip download doesn't contain the icon and loader images, please download the [images](#) and drop that into your folder structure.

You can select the libraries according to your need and click the **Build My Download** button.

## Stable Download

Click the *Stable* button, which leads directly to a ZIP file containing the CSS and JQuery files, for the latest version of jQuery mobile library. Extract the ZIP file contents to a jQuery mobile directory.

This version contains all files including all dependencies, a large collection of demos, and even the library's unit test suite. This version is helpful to getting started.

## Download jQuery Library from CDNs

A CDN (Content Delivery Network) is a network of servers designed to serve files to the users. If you use a CDN link in your webpage, it moves the responsibility of hosting files from your own servers to a series of external ones. This also offers an advantage that if a visitor to your webpage has already downloaded a copy of jQuery mobile from the same CDN, it won't have to be re-downloaded. You can include the following CDN files into the HTML document.

```
//The jQuery Mobile CSS theme file (optional, if you are overriding the default
theme)

<link rel="stylesheet"
href="https://code.jquery.com/mobile/1.4.5/jquery.mobile-1.4.5.min.css">

//The jQuery core JavaScript file

<script src="https://code.jquery.com/jquery-1.11.3.min.js"></script>

//The jQuery Mobile core JavaScript file

<script src="https://code.jquery.com/mobile/1.4.5/jquery.mobile-
1.4.5.min.js"></script>
```

We are using the CDN versions of the library throughout this tutorial. We use AMPPS (AMPPS is a WAMP, MAMP and LAMP stack of Apache, MySQL, MongoDB, PHP, Perl & Python) server to execute all our examples.

### Example

Following is a simple example of jQuery Mobile.

```
<!DOCTYPE html>
<html>
<head>
    <meta name="viewport" content="width=device-width, initial-scale=1">
    <link rel="stylesheet"
href="https://code.jquery.com/mobile/1.4.5/jquery.mobile-1.4.5.min.css">
    <script src="https://code.jquery.com/jquery-1.11.3.min.js"></script>
    <script src="https://code.jquery.com/mobile/1.4.5/jquery.mobile-
1.4.5.min.js"></script>
</head>
<body>
    <div data-role="page" id="pageone">
        <div data-role="header">
            <h1>Header Text</h1>
```

```

</div>

<div data-role="main" class="ui-content">
    <h2>Welcome to TutorialsPoint</h2>
</div>

<div data-role="footer">
    <h1>Footer Text</h1>
</div>
</div>
</body>
</html>

```

Details of the above code are:

- This code is specified inside the head element.

```
<meta name="viewport" content="width=device-width, initial-scale=1">
```

- The viewport is used to specify (by the browser) to display the page zoom level and dimension.
- content="width=device-width" is used to set the pixel width of the page or screen device.
- initial-scale=1 sets the initial zoom level, when the page is loaded for the first time.

- Include the following CDNs

```

<link rel="stylesheet"
      href="https://code.jquery.com/mobile/1.4.5/jquery.mobile-1.4.5.min.css">
<script src="https://code.jquery.com/jquery-1.11.3.min.js"></script>
<script src="https://code.jquery.com/mobile/1.4.5/jquery.mobile-
1.4.5.min.js"></script>

```

- Content inside the <body> tag is a page displayed in the browser.

```

<div data-role="page">
    ...
</div>

```

- *data-role="header"* creates the header at the top of the page.

- *data-role="main"* is used to define the content of the page.
- *data-role="footer"* creates the footer at the bottom of the page.
- *class="ui-content"* includes padding and margin inside the page content.

## Output

Let's carry out the following steps to see how the above code works:

- Save the above html code as **simple\_example.html** file in your server root folder.
- Open this HTML file as [http://localhost/simple\\_example.html](http://localhost/simple_example.html) and the following output will be displayed.

**Header Text**

**Welcome to TutorialsPoint**

**Footer Text**

### 3. jQuery Mobile – Pages

The user can interact with jQuery Mobile pages, which groups the content into logical views and page views. Page view can be animated using page transitions. Multiple pages can be created using HTML document and therefore, there is no need of requesting the content from the server.

Following table demonstrates the types of pages in detail.

| Sr. No. | Types & Description  |
|---------|--|
| 1       | <b><u>Single Page</u></b><br>A single page is created in HTML document using a standard way of writing a template.   |
| 2       | <b><u>Multi-Page Template</u></b><br>Multiple pages can be included in the single HTML document, which loads together by adding multiple divs with <i>data-role="page"</i> . |
| 3       | <b><u>Dialogs Page</u></b><br>Modal dialogs open content in an interactive overlay above the page.   |

#### Single Page

---

##### Description

A standard format is used to create pages many times without changing it. The following standard format is used to create a single page in the jQuery mobile.

```
<div data-role="page">
    <div data-role="header">
        <h1>Page Title</h1>
    </div>

    <div role="main" class="ui-content">
        <p>Page content goes here.</p>
    </div>

    <div data-role="footer">
        <h4>Page Footer</h4>
    </div>
```

```
</div>
```

## Example

The following example demonstrates the use of Single Page in the jQuery Mobile.

```
<!DOCTYPE html>
<html>
<head>
    <meta name="viewport" content="width=device-width, initial-scale=1">
    <link rel="stylesheet"
    href="https://code.jquery.com/mobile/1.4.5/jquery.mobile-1.4.5.min.css">
    <script src="https://code.jquery.com/jquery-1.11.3.min.js"></script>
    <script src="https://code.jquery.com/mobile/1.4.5/jquery.mobile-
1.4.5.min.js"></script>
</head>
<body>
<div data-role="page" id="pageone">
    <div data-role="header">
        <h1>Single Page</h1>
    </div>

    <div data-role="main" class="ui-content">
        <h2>Welcome to TutorialsPoint</h2>
        <p>Lorem Ipsum is simply dummy text of the printing and typesetting industry. Lorem Ipsum has been the industry's standard dummy text ever since the 1500s, when an unknown printer took a galley of type and scrambled it to make a type specimen book.</p>

        <p>It is a long established fact that a reader will be distracted by the readable content of a page when looking at its layout. The point of using Lorem Ipsum is that it has a more-or-less normal distribution of letters, as opposed to using 'Content here, content here', making it look like readable English. Many desktop publishing packages and web page editors now use Lorem Ipsum as their default model text, and a search for 'lorem ipsum' will uncover many web sites still in their infancy</p>
    </div>

    <div data-role="footer">
        <h1>Footer Text</h1>
    </div>
</div>
```

```
</div>
</body>
</html>
```

## Output

Let's carry out the following steps to see how the above code works:

- Save above html code as **single\_page.html** file in your server root folder.
- Open this HTML file as [http://localhost/single\\_page.html](http://localhost/single_page.html) and the following output will be displayed.

# Single Page

## Welcome to TutorialsPoint

Lorem Ipsum is simply dummy text of the printing and typesetting industry. Lorem Ipsum has been the industry's standard dummy text ever since the 1500s, when an unknown printer took a galley of type and scrambled it to make a type specimen book.

It is a long established fact that a reader will be distracted by the readable content of a page when looking at its layout. The point of using Lorem Ipsum is that it has a more-or-less normal distribution of letters, as opposed to using 'Content here, content here', making it look like readable English. Many desktop publishing packages and web page editors now use Lorem Ipsum as their default model text, and a search for 'lorem ipsum' will uncover many web sites still in their infancy

## Footer Text

## Multi-Page Template

---

### Description

Multiple pages can be included in the single jQuery mobile document which loads together by adding multiple divs with the attribute `data-role="page"`. The div with `data-role="page"` should consist of unique id to link internally between the pages.

### Example

Following example demonstrates the use of multi page in the jQuery Mobile.

```
<!DOCTYPE html>
<html>
<head>
    <meta name="viewport" content="width=device-width, initial-scale=1">
    <link rel="stylesheet"
    href="https://code.jquery.com/mobile/1.4.5/jquery.mobile-1.4.5.min.css">
    <script src="https://code.jquery.com/jquery-1.11.3.min.js"></script>
    <script src="https://code.jquery.com/mobile/1.4.5/jquery.mobile-
1.4.5.min.js"></script>
</head>
<body>
<div data-role="page" id="page1">
    <div data-role="header">
        <h2>Header</h2>
    </div>
    <div role="main" class="ui-content">
        <p>Lorem Ipsum is simply dummy text of the printing and typesetting
industry.</p>
        <p> For more information <a href="#page2">click here</a></p>
    </div>
    <div data-role="footer">
        <h4>Footer</h4>
    </div>
</div>
<div data-role="page" id="page2">
    <div data-role="header">
        <h1>Header Text</h1>
    </div>
    <div role="main" class="ui-content">
```

```

<p>Lorem Ipsum is simply dummy text of the printing and typesetting industry. Lorem Ipsum has been the industry's standard dummy text ever since the 1500s.</p>
<p><a href="#page1">Back to previous page</a></p>
</div>
<div data-role="footer">
    <h4>Footer Text</h4>
</div>
</div>
</body>
</html>

```

## Output

Let's carry out the following steps to see how the above code works:

- Save the above html code as **multiple\_page.html** file in your server root folder.
- Open this HTML file as [http://localhost/multiple\\_page.html](http://localhost/multiple_page.html) and the following output will be displayed.

## Header

Lorem Ipsum is simply dummy text of the printing and typesetting industry.

For more information [click here](#)

## Footer

## Header Text

Lorem Ipsum is simply dummy text of the printing and typesetting industry. Lorem Ipsum has been the industry's standard dummy text ever since the 1500s.

[Back to previous page](#)

## Footer Text

—

## Dialogs Page

---

### Description

Modal dialogs open content in an interactive overlay above the page.

Following table demonstrates the types of dialogs pages in detail.

| Sr. No. | Types & Description   |
|---------|---|
| 1       | <b><u>Basic Dialogs</u></b><br>Include attribute <code>data-dialog="true"</code> to any page to display a dialog.                                       |
| 2       | <b><u>Transitions</u></b><br>The transition plugin provides a simple transition effect.   |
| 3       | <b><u>Closing Dialogs</u></b><br>The closing button can be set at the right side by adding <code>data-close-btn="right"</code> to the dialog container. |
| 4       | <b><u>Chaining Dialogs</u></b><br>The <code>data-rel="back"</code> attribute is used to navigate to go back to the previous dialog.                     |
| 5       | <b><u>Styling &amp; theming</u></b><br>Different styling and theming can be applied to a dialog.  |
| 6       | <b><u>Dialog width and margins</u></b><br>By default, the dialog width and margin is set.   |

## Basic Dialogs

---

### Description

Include the attribute `data-dialog="true"` to any page to display a dialog. By default, the dialog box will also have a close button, if the dialog header is included. When the `dialog` attribute is included, then the framework adds styles to it, i.e. rounded corner, margin, etc.

## Example

The following example demonstrates the use of basic dialogs in the jQuery Mobile.

```
<!DOCTYPE html>
<html>
<head>
    <meta name="viewport" content="width=device-width, initial-scale=1">
    <link rel="stylesheet"
    href="https://code.jquery.com/mobile/1.4.5/jquery.mobile-1.4.5.min.css">
    <script src="https://code.jquery.com/jquery-1.11.3.min.js"></script>
    <script src="https://code.jquery.com/mobile/1.4.5/jquery.mobile-
1.4.5.min.js"></script>
</head>
<body>
<div data-role="page" id="page1">
    <div data-role="header">
        <h2>Header</h2>
    </div>

    <div role="main" class="ui-content">
        <p><a href="#page2" class="ui-shadow ui-btn ui-corner-all ui-btn-
inline">Open dialog</a></p>
    </div>

    <div data-role="footer">
        <h4>Footer</h4>
    </div>
</div>

<div data-role="page" data-dialog="true" id="page2">
    <div data-role="header">
        <h1>Dialogs</h1>
    </div>

    <div role="main" class="ui-content">
        <p>Lorem Ipsum is simply dummy text of the printing and typesetting
industry. Lorem Ipsum has been the industry's standard dummy text ever since
the 1500s.</p>
        <p><a href="#page1" class="ui-btn ui-corner-all">Back</a></p>
    </div>
</div>
```

```

</div>
</div>
</body>
</html>
```

## Output

Let's carry out the following steps to see how the above code works:

- Save the above html code as **basic\_dialogs.html** file in your server root folder.
- Open this HTML file as [http://localhost/basic\\_dialogs.html](http://localhost/basic_dialogs.html) and the following output will be displayed.

## Header

[Open dialog](#)

## Footer

# Dialogs

Lorem Ipsum is simply dummy text of the printing and typesetting industry. Lorem Ipsum has been the industry's standard dummy text ever since the 1500s.

[Back](#)

## Transitions Dialogs

---

### Description

The transition plugin provides a simple transition effect. The *data-transition* attribute is used to apply a different transition on the dialog as per choice.

### Example

The following example demonstrates the use of transition in the jQuery Mobile.

```

<!DOCTYPE html>
<html>
<head>
    <meta name="viewport" content="width=device-width, initial-scale=1">
```

```
<link rel="stylesheet"
href="https://code.jquery.com/mobile/1.4.5/jquery.mobile-1.4.5.min.css">
<script src="https://code.jquery.com/jquery-1.11.3.min.js"></script>
<script src="https://code.jquery.com/mobile/1.4.5/jquery.mobile-
1.4.5.min.js"></script>
</head>
<body>
<div data-role="page" id="page1">
<div data-role="header">
<h2>Header</h2>
</div>

<div role="main" class="ui-content">
<a href="#page2" class="ui-shadow ui-btn ui-corner-all ui-btn-inline"
data-transition="slidedown">SlideDown Dialog</a>
<a href="#page3" class="ui-shadow ui-btn ui-corner-all ui-btn-inline"
data-transition="flip">Flip Dialog</a>
</div>

<div data-role="footer">
<h4>Footer</h4>
</div>
</div>

<div data-role="page" data-dialog="true" data-close-btn="right" id="page2">
<div data-role="header">
<h1>Dialogs</h1>
</div>

<div role="main" class="ui-content">
<p>Lorem Ipsum is simply dummy text of the printing and typesetting
industry. Lorem Ipsum has been the industry's standard dummy text ever since
the 1500s.</p>
<p><a href="#page1" class="ui-btn ui-corner-all">Back</a></p>
</div>
</div>

<div data-role="page" data-dialog="true" data-close-btn="none" id="page3">
<div data-role="header">
<h1>Dialogs</h1>
```

```

</div>

<div role="main" class="ui-content">
    <p>Lorem Ipsum is simply dummy text of the printing and typesetting industry. Lorem Ipsum has been the industry's standard dummy text ever since the 1500s.</p>
    <p><a href="#page1" class="ui-btn ui-corner-all">Back</a></p>
</div>
</div>
</body>
</html>

```

## Output

Let's carry out the following steps to see how the above code works:

- Save the above html code as **transition.html** file in your server root folder.
- Open this HTML file as <http://localhost/transition.html> and the following output will be displayed.

## Header

[SlideDown Dialog](#) [Flip Dialog](#)

## Footer

# Dialogs

Lorem Ipsum is simply dummy text of the printing and typesetting industry. Lorem Ipsum has been the industry's standard dummy text ever since the 1500s.

[Back](#)

# Dialogs

Lorem Ipsum is simply dummy text of the printing and typesetting industry. Lorem Ipsum has been the industry's standard dummy text ever since the 1500s.

[Back](#)

## Closing Dialogs

---

### Description

The closing button can be set at the right side by adding `data-close-btn="right"` to the dialog container. By default, the close button is present on the left in the dialog box, when the attribute `dialog` is included. If you don't need the close button in the dialog box, then you can add `data-close-btn="none"` to the dialog container.

### Setting the Close Button Text

- Through the dialog's data-attribute, the close button text can be set.
- It is an important accessibility feature as the close button text can be modified for translating into different languages.
- By default, it displays an icon-only button.
- The text is invisible on the screen but the screen reader can read it.
- The data-attribute `data-close-btn-text` is set to configure the text from your markup.
- The property `$.mobile.dialog.prototype.options.closeBtnText` can be set to a particular string and you can configure for all dialogs by binding to the mobileinit event.

### Example

The following example demonstrates the use of closing dialogs in the jQuery Mobile.

```
<!DOCTYPE html>
<html>
<head>
    <meta name="viewport" content="width=device-width, initial-scale=1">
    <link rel="stylesheet"
    href="https://code.jquery.com/mobile/1.4.5/jquery.mobile-1.4.5.min.css">
    <script src="https://code.jquery.com/jquery-1.11.3.min.js"></script>
    <script src="https://code.jquery.com/mobile/1.4.5/jquery.mobile-
    1.4.5.min.js"></script>
</head>
<body>
<div data-role="page" id="page1">
    <div data-role="header">
        <h2>Header</h2>
    </div>
```

```
<div role="main" class="ui-content">
    <a href="#page2" class="ui-shadow ui-btn ui-corner-all ui-btn-inline">Close Button</a>
    <a href="#page3" class="ui-shadow ui-btn ui-corner-all ui-btn-inline">No Close Button</a>
</div>

<div data-role="footer">
    <h4>Footer</h4>
</div>
</div>

<div data-role="page" data-dialog="true" data-close-btn="right" id="page2">
    <div data-role="header">
        <h1>Dialogs</h1>
    </div>

    <div role="main" class="ui-content">
        <p>Lorem Ipsum is simply dummy text of the printing and typesetting industry. Lorem Ipsum has been the industry's standard dummy text ever since the 1500s.</p>
        <p><a href="#page1" class="ui-btn ui-corner-all">Back</a></p>
    </div>
</div>

<div data-role="page" data-dialog="true" data-close-btn="none" id="page3">
    <div data-role="header">
        <h1>Dialogs</h1>
    </div>

    <div role="main" class="ui-content">
        <p>Lorem Ipsum is simply dummy text of the printing and typesetting industry. Lorem Ipsum has been the industry's standard dummy text ever since the 1500s.</p>
        <p><a href="#page1" class="ui-btn ui-corner-all">Back</a></p>
    </div>
</div>
</body>
```

```
</html>
```

## Output

Let's carry out the following steps to see how the above code works:

- Save the above html code as **closing\_dialogs.html** file in your server root folder.
- Open this HTML file as [http://localhost/closing\\_dialogs.html](http://localhost/closing_dialogs.html) and the following output will be displayed.

## Header

[Close Button](#) [No Close Button](#)

## Footer

# Dialogs

Lorem Ipsum is simply dummy text of the printing and typesetting industry. Lorem Ipsum has been the industry's standard dummy text ever since the 1500s.

[Back](#)

# Dialogs

Lorem Ipsum is simply dummy text of the printing and typesetting industry. Lorem Ipsum has been the industry's standard dummy text ever since the 1500s.

[Back](#)

## Chaining Dialogs

---

### Description

The **data-rel="back"** attribute is used to navigate to go back to the previous dialog. When one dialog is chained to the second dialog box and the second dialog is chained to the third, then using the **data-rel="back"** on the third dialog you can navigate to the second dialog box.

### Example

Following example demonstrates the use of chaining dialogs in jQuery Mobile.

```
<!DOCTYPE html>
<html>
<head>
    <meta name="viewport" content="width=device-width, initial-scale=1">
    <link rel="stylesheet"
    href="https://code.jquery.com/mobile/1.4.5/jquery.mobile-1.4.5.min.css">
    <script src="https://code.jquery.com/jquery-1.11.3.min.js"></script>
    <script src="https://code.jquery.com/mobile/1.4.5/jquery.mobile-
1.4.5.min.js"></script>
</head>
<body>
<div data-role="page" id="page1">
    <div data-role="header">
        <h2>Header</h2>
    </div>

    <div role="main" class="ui-content">
        <a href="#page2" class="ui-shadow ui-btn ui-corner-all ui-btn-
inline">Open Dialog</a>
    </div>

    <div data-role="footer">
        <h4>Footer</h4>
    </div>
</div>

<div data-role="page" data-dialog="true" id="page2">
    <div data-role="header">
        <h1>First Dialogs</h1>
    </div>

    <div role="main" class="ui-content">
        <p>Lorem Ipsum is simply dummy text of the printing and typesetting
industry. Lorem Ipsum has been the industry's standard dummy text ever since
the 1500s.</p>
        <a href="#page3" class="ui-shadow ui-btn ui-corner-all ui-btn-
inline">Open Second Dialog</a>
    </div>
</div>
```

```
<div data-role="page" data-dialog="true" data-close-btn="none" id="page3">
    <div data-role="header">
        <h1>Second Dialogs</h1>
    </div>

    <div role="main" class="ui-content">
        <p>Lorem Ipsum is simply dummy text of the printing and typesetting industry. Lorem Ipsum has been the industry's standard dummy text ever since the 1500s.</p>
        <a href="#page1" class="ui-btn ui-corner-all" data-rel="back">Back to First Dialog </a>
    </div>
</div>
</body>
</html>
```

## Output

Let's carry out the following steps to see how the above code works:

- Save the above html code as **chaining\_dialogs.html** file in your server root folder.
- Open this HTML file as [http://localhost/chaining\\_dialogs.html](http://localhost/chaining_dialogs.html) and the following output will be displayed.

## Header

[Open Dialog](#)

Footer

## First Dialogs

Lorem Ipsum is simply dummy text of the printing and typesetting industry. Lorem Ipsum has been the industry's standard dummy text ever since the 1500s.

[Open Second Dialog](#)

## Second Dialogs

Lorem Ipsum is simply dummy text of the printing and typesetting industry. Lorem Ipsum has been the industry's standard dummy text ever since the 1500s.

[Back to First Dialog](#)

## Styling & Theming Dialogs

---

### Description

Different styling and theming can be applied to a dialog. Following are the ways to apply themes and styles to a dialog:

- Themes can be included in the dialog by adding the attribute data-theme to any containers such as header, footer, or content.
- Include data-corners="false" to remove the rounded corner of the dialog as by default the corners are set as rounded.
- data-overlay-theme attribute sets the overlay to any pattern letter.
- Multiple button can also be used in the dialogs. The dialog can be fixed to the top by including {margin-top:0} to the class specified for the page **.ui-dialog.my-dialog .ui-dialog-contain**

## Example

Following example demonstrates the use of styling and theming dialogs in the jQuery Mobile.

```
<!DOCTYPE html>
<html>
<head>
    <meta name="viewport" content="width=device-width, initial-scale=1">
    <link rel="stylesheet"
    href="https://code.jquery.com/mobile/1.4.5/jquery.mobile-1.4.5.min.css">
    <script src="https://code.jquery.com/jquery-1.11.3.min.js"></script>
    <script src="https://code.jquery.com/mobile/1.4.5/jquery.mobile-
1.4.5.min.js"></script>
    <style>
        .ui-dialog.dialog-actionsheet .ui-dialog-contain {
            margin-top: 0;
        }
    </style>
</head>
<body>
    <div data-role="page" id="page1">
        <div data-role="header" data-theme="b">
            <h2>Header</h2>
        </div>

        <div role="main" class="ui-content">
            <a href="#page2" class="ui-btn ui-corner-all ui-btn-inline">Open
Dialog</a>
            <a href="#page3" class="ui-btn ui-corner-all ui-btn-inline">Photos
Dialog</a>
        </div>

        <div data-role="footer">
            <h4>Footer</h4>
        </div>
    </div>

    <div data-role="page" data-dialog="true" id="page2" data-corners="false"
data-overlay-theme="b">
```

```

<div data-role="header">
    <h1>First Dialogs</h1>
</div>

<div role="main" class="ui-content">
    <p>Lorem Ipsum is simply dummy text of the printing and typesetting industry. Lorem Ipsum has been the industry's standard dummy text ever since the 1500s.</p>
    <a data-rel="back" class="ui-btn ui-corner-all ui-btn-inline">Back</a>
</div>
</div>

<div data-role="page" class="dialog-actionsheet" data-dialog="true" id="page3">
    <div role="main" class="ui-content">
        <h3>Share Photos</h3>
        <a href="#page4" data-transition="slidedown" class="ui-btn ui-shadow ui-corner-all ui-btn-b">Email</a>
        <a href="#page4" data-transition="slidedown" class="ui-btn ui-shadow ui-corner-all ui-btn-b">Share on Facebook</a>
        <a href="#page4" data-transition="slidedown" class="ui-btn ui-shadow ui-corner-all ui-btn-b">Tweet photo</a>
        <a data-rel="back" class="ui-btn ui-shadow ui-corner-all ui-btn-a">Cancel</a>
    </div>
</div>

<div data-role="page" class="dialog-actionsheet" data-dialog="true" id="page4">
    <div role="main" class="ui-content">
        <h3>Photos uploaded successfully</h3>
        <a href="#page1" class="ui-btn ui-shadow ui-corner-all ui-btn-a">View photo page</a>
        <a href="#page1" class="ui-btn ui-shadow ui-corner-all ui-btn-a">Done</a>
    </div>
</div>
</body>
</html>

```

## Output

Let's carry out the following steps to see how the above code works:

- Save the above html code as **style\_theming.html** file in your server root folder.
- Open this HTML file as [http://localhost/style\\_theming.html](http://localhost/style_theming.html) and the following output will be displayed.

## Header

[Open Dialog](#) [Photos Dialog](#)

## Footer

# First Dialogs

Lorem Ipsum is simply dummy text of the printing and typesetting industry. Lorem Ipsum has been the industry's standard dummy text ever since the 1500s.

[Back](#)

## Share Photos

[Email](#) [Share on Facebook](#) [Tweet photo](#) [Cancel](#)

**Photos uploaded successfully**

[View photo page](#) [Done](#)

## Dialog Width and Margins

By default, the dialog width and margin is set in the jQuery mobile. The default width of dialog is 92.5%. On the larger screens, the dialog top margin is set larger, i.e. 10% top margin but it gets collapsed on the small screen to a small margin. This style can be overriden in your stylesheet when needed.

```
.ui-dialog-contain {
    width: 92.5%;
    max-width: 500px;
    margin: 10% auto 15px auto;
    padding: 0;
    position: relative;
    top: -15px;
```

```
}
```

## Conventions, Not Requirements

- The data-role attribute elements such as the header, footer, page, and content are used to provide the basic format and structure of a page.
- For single page documents, the page wrapper was required for auto-initialization is set as optional.
- The structural element can be excluded for a webpage with custom layout.
- To manage pages, the page wrapper is injected by the framework when it is not included by the markup.

## Prefetching Pages

Including the attribute *data-prefetch*, we can prefetch pages into the DOM in the single page templates. For more information [click here](#).

### Prefetching Pages

---

#### Description

Including the attribute *data-prefetch*, we can prefetch pages into the DOM in the single page templates. It helps to link those points to the page. After the primary page is loaded, the target page is loaded in the background and the **pagecreate** event is triggered.

Using the pagecontainer widget's *load()* method, we can prefetch a page alternatively.

```
$( ":mobile-pagecontainer" ).pagecontainer( "load", pageTitle, { showLoadMsg: false } );
```

#### Example

The following example demonstrates the use of Prefetching Pages in the jQuery Mobile.

```
<!DOCTYPE html>
<html>
<head>
  <meta name="viewport" content="width=device-width, initial-scale=1">
  <link rel="stylesheet"
  href="https://code.jquery.com/mobile/1.4.5/jquery.mobile-1.4.5.min.css">
  <script src="https://code.jquery.com/jquery-1.11.3.min.js"></script>
  <script src="https://code.jquery.com/mobile/1.4.5/jquery.mobile-
  1.4.5.min.js"></script>
</head>
```

```

<body>
<div data-role="page" id="page1">
    <div data-role="header">
        <h2>Header</h2>
    </div>

    <div data-role="main" class="ui-content">
        <h2>Welcome to Tutorialspoint!!! </h2>
        <a href="#page2" data-prefetch="true" class="ui-btn ui-btn-inline">Next
Page</a>
    </div>

    <div data-role="footer">
        <h2>Footer</h2>
    </div>
</div>

<div data-role="page" data-dialog="true" id="page2">
    <div data-role="header">
        <h2>Header</h2>
    </div>

    <div data-role="main" class="ui-content">
        <p>Click the link to go page one.</p>
        <a href="#page1" class="ui-btn">Back to Previous Page</a>
    </div>

    <div data-role="footer">
        <h2>Footer</h2>
    </div>
</div>
</body>
</html>

```

## Output

Let's carry out the following steps to see how the above code works:

- Save the above html code as **prefetching\_page.html** file in your server root folder.
- Open this HTML file as [http://localhost/prefetching\\_page.html](http://localhost/prefetching_page.html) and the following output will be displayed.

## Header

# Welcome to TutorialsPoint!!!

[Next Page](#)

## Footer

## Header

Click the link to go page one.

[Back to Previous Page](#)

## Footer

-

## DOM Cache

When the browser memory gets full in DOM, then it slows down the mobile browser or might crash due to loading of multiple pages. There is a simple method to keep the DOM tidy:

- When a page is loaded via ajax, then it indicates to remove the page from DOM when you redirect to another page.
- The previous page which you have visited can be retrieved from the cache when you revisit it.
- Instead of removing the pages, you can tell jQuery mobile to keep it in DOM by using the following line:

```
$.mobile.page.prototype.options.domCache = true;
```

- Set the `domCache` option as true on the `page` plugin to keep all the pages in the DOM, which was visited previously.

```
pageContainerElement.page({ domCache: true });
```

## 4. jQuery Mobile – Icons

jQuery Mobile provides a set of built-in icons, which can be used with buttons, listview buttons which will make the buttons more attractive.

Following table lists down some of the icons used in the jQuery Mobile framework.

| Sr. No. | Icon Area & Description   |
|---------|---|
| 1       | <b><u>Icon Set</u></b><br>It sets the icon in the button.   |
| 2       | <b><u>Positioning Icons</u></b><br>It specifies the position of the icon in the button.                     |
| 3       | <b><u>Icon-only</u></b><br>It displays only an icon in the button.  |
| 4       | <b><u>Icon shadow</u></b><br>It adds an icon shadow in your button.   |
| 5       | <b><u>Removing Circle</u></b><br>It removes the grey circle around the icon.                                |
| 6       | <b><u>Black and White Icons</u></b><br>It changes the color of icons to black or white.                     |
| 7       | <b><u>Combining alt and nodisc</u></b><br>It combines the <i>alt</i> and <i>nodisc</i> classes to the icon. |

### Icon Set

---

#### Description

You can set the icon in the button using ***ui-icon*** class and ***ui-btn-icon-pos\_name*** class for specifying the position for icon.

## Example

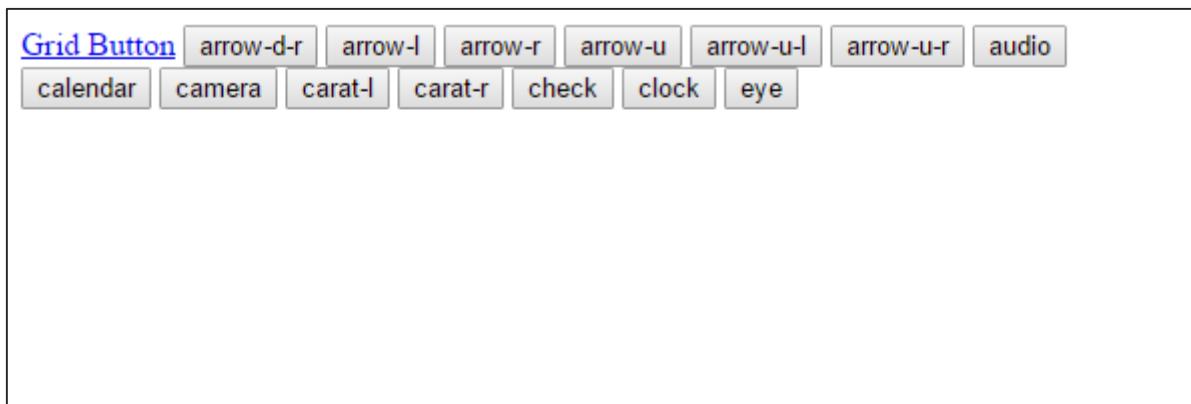
Following example describes the use of icon setting in the jQuery Mobile Framework.

```
<!DOCTYPE html>
<head>
<meta name="viewport" content="width=device-width, initial-scale=1">
<link rel="stylesheet"
href="https://code.jquery.com/mobile/1.4.5/jquery.mobile-1.4.5.min.css">
<script src="https://code.jquery.com/jquery-1.11.3.min.js"></script>
<script src="https://code.jquery.com/mobile/1.4.5/jquery.mobile-
1.4.5.min.js"></script>
</head>
<body>
<div data-role="main" class="ui-content">
    <a href="#" class="ui-btn ui-btn-icon-left ui-icon-grid">Grid Button</a>
    <button class="ui-btn ui-btn-icon-left ui-icon-arrow-d-r">arrow-d-r</button>
    <button class="ui-btn ui-btn-icon-left ui-icon-arrow-l">arrow-l</button>
    <button class="ui-btn ui-btn-icon-left ui-icon-arrow-r">arrow-r</button>
    <button class="ui-btn ui-btn-icon-left ui-icon-arrow-u">arrow-u</button>
    <button class="ui-btn ui-btn-icon-left ui-icon-arrow-u-l">arrow-u-l</button>
    <button class="ui-btn ui-btn-icon-left ui-icon-arrow-u-r">arrow-u-r</button>
    <button class="ui-btn ui-btn-icon-left ui-icon-audio">audio</button>
    <button class="ui-btn ui-btn-icon-left ui-icon-calendar">calendar</button>
    <button class="ui-btn ui-btn-icon-left ui-icon-camera">camera</button>
    <button class="ui-btn ui-btn-icon-left ui-icon-carat-l">carat-l</button>
    <button class="ui-btn ui-btn-icon-left ui-icon-carat-r">carat-r</button>
    <button class="ui-btn ui-shadow ui-corner-all ui-btn-icon-left ui-icon-
check">check</button>
    <button class="ui-btn ui-shadow ui-corner-all ui-btn-icon-left ui-icon-
clock">clock</button>
    <button class="ui-btn ui-shadow ui-corner-all ui-btn-icon-left ui-icon-
eye">eye</button>
</div>
</body>
</html>
```

## Output

Let's carry out the following steps to see how the above code works:

- Save above html code as **icons\_icon\_set.html** file in your server root folder.
- Open this HTML file as [http://localhost/icons\\_icon\\_set.html](http://localhost/icons_icon_set.html) and the following output will be displayed.



## Icon Positioning

---

### Description

You can determine the position of the icon (top, right, left, bottom in the button) using the ***ui-btn-icon-[value]*** class.

### Example

Following example describes the use of icon positioning in the jQuery Mobile Framework.

```
<!DOCTYPE html>
<head>
<title>Icon Setting</title>
<meta name="viewport" content="width=device-width, initial-scale=1">
<link rel="stylesheet"
href="https://code.jquery.com/mobile/1.4.5/jquery.mobile-1.4.5.min.css">
<script src="https://code.jquery.com/jquery-1.11.3.min.js"></script>
<script src="https://code.jquery.com/mobile/1.4.5/jquery.mobile-
1.4.5.min.js"></script>
</head>
<body>
<div data-role="main" class="ui-content">
<a href="" class="ui-btn ui-icon-home ui-btn-icon-left">Left</a>
<a href="" class="ui-btn ui-icon-home ui-btn-icon-right">Right</a>
```

```

<a href="" class="ui-btn ui-icon-home ui-btn-icon-top">Top</a>
<a href="" class="ui-btn ui-icon-home ui-btn-icon-bottom">Bottom</a>
</div>
</body>
</html>

```

## Output

Let's carry out the following steps to see how the above code works:

- Save the above html code as **icons\_positioning.html** file in your server root folder.
- Open this HTML file as [http://localhost/icons\\_positioning.html](http://localhost/icons_positioning.html) and the following output will be displayed.

[Left](#) [Right](#) [Top](#) [Bottom](#)

## Icon Only

---

### Description

You can display only an icon in the button using the **ui-btn-icon-notext** class.

### Example

Following example describes the use of *displaying only icon* in the jQuery Mobile Framework.

```

<!DOCTYPE html>
<head>
    <title>Displaying only Icon</title>
    <meta name="viewport" content="width=device-width, initial-scale=1">
    <link rel="stylesheet"
    href="https://code.jquery.com/mobile/1.4.5/jquery.mobile-1.4.5.min.css">
    <script src="https://code.jquery.com/jquery-1.11.3.min.js"></script>

```

```

<script src="https://code.jquery.com/mobile/1.4.5/jquery.mobile-
1.4.5.min.js"></script>

</head>

<body>

<div data-role="page" id="page1">

    <div data-role="header">
        <h2>Header</h2>
    </div>

    <div data-role="main" class="ui-content">
        <a href="" class="ui-btn ui-icon-home ui-btn-icon-notext">Delete</a>
    </div>

    <div data-role="footer">
        <h2>Footer</h2>
    </div>
</div>
</body>
</html>

```

## Output

Let's carry out the following steps to see how the above code works:

- Save the above html code as **icons\_only\_icon.html** file in your server root folder.
- Open this HTML file as [http://localhost/icons\\_only\\_icon.html](http://localhost/icons_only_icon.html) and the following output will be displayed.

**Header**

[Delete](#)

**Footer**

## Icon Shadow

---

### Description

You can add an icon shadow in your button using the ***ui-shadow-icon*** class.

### Example

Following example describes the use of icon shadow in the jQuery Mobile Framework.

```
<!DOCTYPE html>
<head>
    <title>Displaying Icon Shadow</title>
    <meta name="viewport" content="width=device-width, initial-scale=1">
    <link rel="stylesheet"
    href="https://code.jquery.com/mobile/1.4.5/jquery.mobile-1.4.5.min.css">
    <script src="https://code.jquery.com/jquery-1.11.3.min.js"></script>
    <script src="https://code.jquery.com/mobile/1.4.5/jquery.mobile-
1.4.5.min.js"></script>
</head>
<body>
<div data-role="page" id="page1">
    <div data-role="header">
        <h2>Header</h2>
    </div>

    <div data-role="main" class="ui-content">
        <a href="" class="ui-btn ui-shadow-icon ui-shadow ui-corner-all ui-icon-
home ui-btn-icon-notext">Delete</a>
    </div>

    <div data-role="footer">
        <h2>Footer</h2>
    </div>
</div>
</body>
</html>
```

### Output

Let's carry out the following steps to see how above code works:

- Save the above html code as **icons\_shadow\_icon.html** file in your server root folder.
- Open this HTML file as [http://localhost/icons\\_shadow\\_icon.html](http://localhost/icons_shadow_icon.html) and the following output will be displayed.

## Header

[Delete](#)

## Footer

## Removing Icon Circle

---

### Description

You can remove the grey circle around the icon using the **ui-nodisc-icon** class.

### Example

Following example describes the use of removing icon *circle* in the jQuery Mobile Framework.

```
<!DOCTYPE html>
<head>
    <title>Displaying Icon Shadow</title>
    <meta name="viewport" content="width=device-width, initial-scale=1">
    <link rel="stylesheet"
    href="https://code.jquery.com/mobile/1.4.5/jquery.mobile-1.4.5.min.css">
    <script src="https://code.jquery.com/jquery-1.11.3.min.js"></script>
    <script src="https://code.jquery.com/mobile/1.4.5/jquery.mobile-
    1.4.5.min.js"></script>
</head>
<body>
<div data-role="page" id="page1">
    <div data-role="header">
        <h2>Header</h2>
```

```

</div>

<div data-role="main" class="ui-content">
    <a href="" class="ui-nodisc-icon ui-btn ui-shadow ui-corner-all ui-btn-
    inline ui-icon-delete ui-btn-icon-left">Without Grey Circle</a>
</div>

<div data-role="footer">
    <h2>Footer</h2>
</div>
</div>
</body>
</html>

```

## Output

Let's carry out the following steps to see how the above code works:

- Save the above html code as **icons\_removing\_circle.html** file in your server root folder.
- Open this HTML file as [http://localhost/icons\\_removing\\_circle.html](http://localhost/icons_removing_circle.html) and the following output will be displayed.

## Header

[Without Grey Circle](#)

## Footer

## Black and White Icon Sets

---

### Description

You can change the color of an icon to black using the **ui-alt-icon** class. By default, all icons are white.

## Example

Following example describes the use of changing icon color in the jQuery Mobile Framework.

```
<!DOCTYPE html>
<head>
    <title>Black and White Icons</title>
    <meta name="viewport" content="width=device-width, initial-scale=1">
    <link rel="stylesheet"
    href="https://code.jquery.com/mobile/1.4.5/jquery.mobile-1.4.5.min.css">
    <script src="https://code.jquery.com/jquery-1.11.3.min.js"></script>
    <script src="https://code.jquery.com/mobile/1.4.5/jquery.mobile-
1.4.5.min.js"></script>
</head>
<body>
<div data-role="page" id="page1">
    <div data-role="header">
        <h2>Header</h2>
    </div>

    <div data-role="main" class="ui-content">
        <a href="" class="ui-btn ui-btn-inline ui-icon-home ui-btn-icon-left ui-
corner-all">White icon</a>
        <a href="" class="ui-btn ui-btn-inline ui-icon-home ui-btn-icon-left ui-
corner-all ui-alt-icon">Black icon</a>
    </div>

    <div data-role="footer">
        <h2>Footer</h2>
    </div>
</div>
</body>
</html>
```

## Output

Let's carry out the following steps to see how the above code works:

- Save the above html code as **icons\_black\_white.html** file in your server root folder.

- Open this HTML file as `http://localhost/icons_black_white.html` and the following output will be displayed.

## Header

[White icon](#) [Black icon](#)

## Footer

## Combining alt and nodisc

---

### Description

You can combine the **`alt(ui-alt-icon)`** and **`nodisc(ui-nodisc-icon)`** classes.

### Example

Following example describes the use of combining icon alt *and* nodisc classes in the jQuery Mobile Framework.

```
<!DOCTYPE html>
<head>
    <title>Alt and Nodisc Classes</title>
    <meta name="viewport" content="width=device-width, initial-scale=1">
    <link rel="stylesheet"
    href="https://code.jquery.com/mobile/1.4.5/jquery.mobile-1.4.5.min.css">
    <script src="https://code.jquery.com/jquery-1.11.3.min.js"></script>
    <script src="https://code.jquery.com/mobile/1.4.5/jquery.mobile-
    1.4.5.min.js"></script>
</head>
<body>
<div data-role="page" id="page1">
    <div data-role="header">
        <h2>Header</h2>
    </div>

    <div data-role="main" class="ui-content">
```

```
<a href="" class="ui-btn ui-btn-inline ui-icon-home ui-btn-icon-left ui-corner-all ui-alt-icon ui-nodisc-icon">Black icon</a>
</div>

<div data-role="footer">
    <h2>Footer</h2>
</div>
</div>
</body>
</html>
```

## Output

Let's carry out the following steps to see how the above code works:

- Save the above html code as **icons\_combng\_alt\_nodisc.html** file in your server root folder.
- Open this HTML file as [http://localhost/icons\\_combng\\_alt\\_nodisc.html](http://localhost/icons_combng_alt_nodisc.html) and the following output will be displayed.

**Header**

[Black icon](#)

**Footer**

## 5. jQuery Mobile – Transitions

It allows to change the property values, which occurs over a specified duration and alters the behavior of an element from one state to another state, by applying different styles for each state.

Following table lists down some of the page transitions used in the jQuery Mobile framework.

| Sr. No. | Transition & Description   | For Pages                      | For Dialogs                      |
|---------|--|--------------------------------|----------------------------------|
| 1       | <b>fade</b><br>You can make elements fade in and out of visibility.                        | <a href="#">Fade Page</a>      | <a href="#">Fade Dialog</a>      |
| 2       | <b>flip</b><br>Flip the elements from back to front to the next page.                      | <a href="#">Flip Page</a>      | <a href="#">Flip Dialog</a>      |
| 3       | <b>pop</b><br>You can create a popup window.   | <a href="#">Pop Page</a>       | <a href="#">Pop Dialog</a>       |
| 4       | <b>flow</b><br>Display the next page by keeping the current page away.                     | <a href="#">Flow Page</a>      | <a href="#">Flow Dialog</a>      |
| 5       | <b>slide</b><br>You can slide the page from the right to the left.                         | <a href="#">Slide Page</a>     | <a href="#">Slide Dialog</a>     |
| 6       | <b>slidefade</b><br>Slides the page from the right to the left and fades in the next page. | <a href="#">Slidefade Page</a> | <a href="#">Slidefade Dialog</a> |
| 7       | <b>slideup</b><br>Slides the page from bottom to up.                                       | <a href="#">Slideup Page</a>   | <a href="#">Slideup Dialog</a>   |
| 8       | <b>slidedown</b><br>Slides the page from top to bottom.                                    | <a href="#">Slidedown Page</a> | <a href="#">Slidedown Dialog</a> |
| 9       | <b>turn</b><br>You can turn to the next page.  | <a href="#">Turn Page</a>      | <a href="#">Turn Dialog</a>      |
| 10      | <b>none</b><br>You cannot use any transition effect by using this attribute.               | <a href="#">None Page</a>      | <a href="#">None Dialog</a>      |

## Fade Page Transition

### Description

You can make the elements fade in and out of visibility for the pages using the ***data-transition="fade"*** attribute.

### Example

Following example describes the use of fading page in the jQuery Mobile Framework.

```
<!DOCTYPE html>
<head>
    <meta name="viewport" content="width=device-width, initial-scale=1">
    <link rel="stylesheet"
    href="https://code.jquery.com/mobile/1.4.5/jquery.mobile-1.4.5.min.css">
    <script src="https://code.jquery.com/jquery-1.11.3.min.js"></script>
    <script src="https://code.jquery.com/mobile/1.4.5/jquery.mobile-
1.4.5.min.js"></script>
</head>
<body>
<div data-role="page" id="fade_page1">
    <div data-role="header">
        <h2>Header</h2>
    </div>
    <div data-role="main" class="ui-content">
        <p>Welcome to Tutorialspoint!!! Click the link to go page two.</p>
        <a href="#fade_page2" data-transition="fade">Page Two</a>
    </div>

    <div data-role="footer">
        <h2>Footer</h2>
    </div>
</div>

<div data-role="page" id="fade_page2">
    <div data-role="header">
        <h2>Header</h2>
    </div>
```

```

<div data-role="main" class="ui-content">
    <p>Click the link to go page one.</p>
    <a href="#fade_page1">Page One</a>
</div>

<div data-role="footer">
    <h2>Footer</h2>
</div>
</div>
</body>
</html>

```

## Output

Let's carry out the following steps to see how the above code works:

- Save the above html code as **page\_fade.html** file in your server root folder.
- Open this HTML file as [http://localhost/page\\_fade.html](http://localhost/page_fade.html) and the following output will be displayed.

## Header

Welcome to Tutorialspoint!!! Click the link to go page two.

[Page Two](#)

## Footer

## Header

Click the link to go page one.

[Page One](#)

## Footer

## Fade Dialog Transition

### Description

You can make the elements fade in and out of visibility for the dialog box using the ***data-transition="fade"*** attribute.

### Example

Following example describes the use of fading dialog box in the jQuery Mobile Framework.

```
<!DOCTYPE html>
<head>
    <meta name="viewport" content="width=device-width, initial-scale=1">
    <link rel="stylesheet"
    href="https://code.jquery.com/mobile/1.4.5/jquery.mobile-1.4.5.min.css">
    <script src="https://code.jquery.com/jquery-1.11.3.min.js"></script>
    <script src="https://code.jquery.com/mobile/1.4.5/jquery.mobile-
1.4.5.min.js"></script>
</head>
<body>
<div data-role="page" id="page1">
    <div data-role="header">
        <h2>Header</h2>
    </div>
    <div data-role="main" class="ui-content">
        <p>Welcome to Tutorialspoint!!! Click the link to see dialog box.</p>
        <a href="#page2" data-transition="fade">Page Two</a>
    </div>
    <div data-role="footer">
        <h2>Footer</h2>
    </div>
</div>

<div data-role="page" data-dialog="true" id="page2">
    <div data-role="header">
        <h2>This is Dialog box</h2>
    </div>

    <div data-role="main" class="ui-content">
```

```

<p>Click the link to go page one.</p>
<a href="#page1">Page One</a>
</div>

<div data-role="footer">
  <h2>Footer</h2>
</div>
</div>
</body>
</html>

```

## Output

Let's carry out the following steps to see how the above code works:

- Save the above html code as **dialog\_fade.html** file in your server root folder.
- Open this HTML file as [http://localhost/dialog\\_fade.html](http://localhost/dialog_fade.html) and the following output will be displayed.

## Header

Welcome to Tutorialspoint!!! Click the link to see dialog box.

[Page Two](#)

## Footer

### This is Dialog box

Click the link to go page one.

[Page One](#)

## Footer

## Flip Page Transition

### Description

You can flip the elements from the back to the front to the next page using the ***data-transition="flip"*** attribute.

### Example

Following example describes the use of page flip in the jQuery Mobile Framework.

```
<!DOCTYPE html>
<head>
    <meta name="viewport" content="width=device-width, initial-scale=1">
    <link rel="stylesheet"
    href="https://code.jquery.com/mobile/1.4.5/jquery.mobile-1.4.5.min.css">
    <script src="https://code.jquery.com/jquery-1.11.3.min.js"></script>
    <script src="https://code.jquery.com/mobile/1.4.5/jquery.mobile-
1.4.5.min.js"></script>
</head>
<body>
<div data-role="page" id="page1">
    <div data-role="header">
        <h2>Header</h2>
    </div>
    <div data-role="main" class="ui-content">
        <p>Welcome to Tutorialspoint!!! Click the link to flip the page.</p>
        <a href="#page2" data-transition="flip">Page Two</a>
    </div>
    <div data-role="footer">
        <h2>Footer</h2>
    </div>
</div>

<div data-role="page" id="page2">
    <div data-role="header">
        <h2>Header</h2>
    </div>

    <div data-role="main" class="ui-content">
```

```

<p>Click the link to go page one.</p>
<a href="#page1">Page One</a>
</div>

<div data-role="footer">
  <h2>Footer</h2>
</div>
</div>
</body>
</html>

```

## Output

Let's carry out the following steps to see how the above code works:

- Save the above html code as **page\_flip.html** file in your server root folder.
- Open this HTML file as [http://localhost/page\\_flip.html](http://localhost/page_flip.html) and the following output will be displayed.

## Header

Welcome to Tutorialspoint!!! Click the link to flip the page.

[Page Two](#)

## Footer

## Header

Click the link to go page one.

[Page One](#)

## Footer

## Flip Dialog Transition

### Description

You can flip the elements from the back to the front to the next page using the dialog box by setting the ***data-transition="flip"*** attribute.

### Example

Following example describes the use of dialog flip in the jQuery Mobile Framework.

```
<!DOCTYPE html>
<head>
    <meta name="viewport" content="width=device-width, initial-scale=1">
    <link rel="stylesheet"
    href="https://code.jquery.com/mobile/1.4.5/jquery.mobile-1.4.5.min.css">
    <script src="https://code.jquery.com/jquery-1.11.3.min.js"></script>
    <script src="https://code.jquery.com/mobile/1.4.5/jquery.mobile-
    1.4.5.min.js"></script>
</head>
<body>
<div data-role="page" id="page1">
    <div data-role="header">
        <h2>Header</h2>
    </div>
    <div data-role="main" class="ui-content">
        <p>Welcome to Tutorialspoint!!! Click the link to flip the page.</p>
        <a href="#page2" data-transition="flip">Page Two</a>
    </div>
    <div data-role="footer">
        <h2>Footer</h2>
    </div>
</div>

<div data-role="page" data-dialog="true" id="page2">
    <div data-role="header">
        <h2>Header</h2>
    </div>

    <div data-role="main" class="ui-content">
```

```

<p>Click the link to go page one.</p>
<a href="#page1">Page One</a>
</div>

<div data-role="footer">
  <h2>Footer</h2>
</div>
</div>
</body>
</html>

```

## Output

Let's carry out the following steps to see how the above code works:

- Save the above html code as **dialog\_flip.html** file in your server root folder.
- Open this HTML file as [http://localhost/dialog\\_flip.html](http://localhost/dialog_flip.html) and the following output will be displayed.

## Page Popup Transition

### Description

Create a popup window for the page using the **data-transition="pop"** attribute.

### Example

Following example describes the use of page popup in the jQuery Mobile Framework.

```

<!DOCTYPE html>
<head>
  <meta name="viewport" content="width=device-width, initial-scale=1">
  <link rel="stylesheet"
  href="https://code.jquery.com/mobile/1.4.5/jquery.mobile-1.4.5.min.css">
  <script src="https://code.jquery.com/jquery-1.11.3.min.js"></script>
  <script src="https://code.jquery.com/mobile/1.4.5/jquery.mobile-
  1.4.5.min.js"></script>
</head>
<body>
<div data-role="page" id="page1">
  <div data-role="header">
    <h2>Header</h2>

```

```

</div>
<div data-role="main" class="ui-content">
    <p>Welcome to Tutorialspoint!!! Click the link to open the popup
    window.</p>
    <a href="#page2" data-transition="pop">Page Two</a>
</div>
<div data-role="footer">
    <h2>Footer</h2>
</div>
</div>

<div data-role="page" id="page2">
    <div data-role="header">
        <h2>Header</h2>
    </div>

    <div data-role="main" class="ui-content">
        <p>Click the link to go page one.</p>
        <a href="#page1">Page One</a>
    </div>

    <div data-role="footer">
        <h2>Footer</h2>
    </div>
</div>
</body>
</html>

```

## Output

Let's carry out the following steps to see how the above code works:

- Save the above html code as **page\_popup.html** file in your server root folder.
- Open this HTML file as [http://localhost/page\\_popup.html](http://localhost/page_popup.html) and the following output will be displayed.

## Header

Welcome to Tutorialspoint!!! Click the link to open the popup window.

[Page Two](#)

## Footer

## Header

Click the link to go page one.

[Page One](#)

## Footer

## Dialog Popup Transition

---

### Description

Create a popup window for the page using the dialog box by setting the **data-transition="pop"** attribute.

### Example

Following example describes the use of dialog popup in the jQuery Mobile Framework.

```
<!DOCTYPE html>
<head>
    <meta name="viewport" content="width=device-width, initial-scale=1">
    <link rel="stylesheet"
    href="https://code.jquery.com/mobile/1.4.5/jquery.mobile-1.4.5.min.css">
    <script src="https://code.jquery.com/jquery-1.11.3.min.js"></script>
    <script src="https://code.jquery.com/mobile/1.4.5/jquery.mobile-
    1.4.5.min.js"></script>
</head>
<body>
<div data-role="page" id="page1">
    <div data-role="header">
        <h2>Header</h2>
```

```

</div>
<div data-role="main" class="ui-content">
<p>Welcome to Tutorialspoint!!! Click the link to open the popup window
using dialog box.</p>
<a href="#page2" data-transition="pop">Page Two</a>
</div>
<div data-role="footer">
<h2>Footer</h2>
</div>
</div>

<div data-role="page" data-dialog="true" id="page2">
<div data-role="header">
<h2>Header</h2>
</div>

<div data-role="main" class="ui-content">
<p>Click the link to go page one.</p>
<a href="#page1">Page One</a>
</div>

<div data-role="footer">
<h2>Footer</h2>
</div>
</div>
</body>
</html>

```

## Output

Let's carry out the following steps to see how the above code works:

- Save the above html code as **dialog\_popup.html** file in your server root folder.
- Open this HTML file as [http://localhost/dialog\\_popup.html](http://localhost/dialog_popup.html) and the following output will be displayed.

## Header

Welcome to Tutorialspoint!!! Click the link to open the popup window using dialog box.

[Page Two](#)

## Footer

## Header

Click the link to go page one.

[Page One](#)

## Footer

## Page Flow Transition

---

### Description

You can display the next page by keeping the current page away by using the **data-transition="flow"** attribute.

### Example

Following example describes the use of page flow in the jQuery Mobile Framework.

```
<!DOCTYPE html>
<head>
    <meta name="viewport" content="width=device-width, initial-scale=1">
    <link rel="stylesheet"
    href="https://code.jquery.com/mobile/1.4.5/jquery.mobile-1.4.5.min.css">
    <script src="https://code.jquery.com/jquery-1.11.3.min.js"></script>
    <script src="https://code.jquery.com/mobile/1.4.5/jquery.mobile-
    1.4.5.min.js"></script>
</head>
<body>
<div data-role="page" id="page1">
    <div data-role="header">
        <h2>Header</h2>
```

```

</div>
<div data-role="main" class="ui-content">
    <p>Welcome to Tutorialspoint!!! Click the link to throw the current page away and comes in with the next page.</p>
    <a href="#page2" data-transition="flow">Page Two</a>
</div>
<div data-role="footer">
    <h2>Footer</h2>
</div>
</div>

<div data-role="page" id="page2">
    <div data-role="header">
        <h2>Header</h2>
    </div>

    <div data-role="main" class="ui-content">
        <p>Click the link to go page one.</p>
        <a href="#page1">Page One</a>
    </div>

    <div data-role="footer">
        <h2>Footer</h2>
    </div>
</div>
</body>
</html>

```

## Output

Let's carry out the following steps to see how the above code works:

- Save the above html code as **page\_flow.html** file in your server root folder.
- Open this HTML file as [http://localhost/page\\_flow.html](http://localhost/page_flow.html) and the following output will be displayed.

## Header

Welcome to Tutorialspoint!!! Click the link to throw the current page away and comes in with the next page.

[Page Two](#)

## Footer

## Header

Click the link to go page one.

[Page One](#)

## Footer

## Dialog Flow Transition

---

### Description

You can display the next page by keeping the current page away using the dialog box by setting the ***data-transition="flow"*** attribute.

### Example

Following example describes the use of dialog flow in the jQuery Mobile Framework.

```
<!DOCTYPE html>
<head>
    <meta name="viewport" content="width=device-width, initial-scale=1">
    <link rel="stylesheet"
    href="https://code.jquery.com/mobile/1.4.5/jquery.mobile-1.4.5.min.css">
    <script src="https://code.jquery.com/jquery-1.11.3.min.js"></script>
    <script src="https://code.jquery.com/mobile/1.4.5/jquery.mobile-
    1.4.5.min.js"></script>
</head>
<body>
<div data-role="page" id="page1">
    <div data-role="header">
        <h2>Header</h2>
```

```
</div>
<div data-role="main" class="ui-content">
    <p>Welcome to Tutorialspoint!!! Click the link to throw the current page away and comes in with the next page using dialog box.</p>
    <a href="#page2" data-transition="flow">Page Two</a>
</div>
<div data-role="footer">
    <h2>Footer</h2>
</div>
</div>

<div data-role="page" data-dialog="true" id="page2">
    <div data-role="header">
        <h2>Header</h2>
    </div>

    <div data-role="main" class="ui-content">
        <p>Click the link to go page one.</p>
        <a href="#page1">Page One</a>
    </div>

    <div data-role="footer">
        <h2>Footer</h2>
    </div>
</div>
</body>
</html>
```

## Output

Let's carry out the following steps to see how the above code works:

- Save the above html code as **dialog\_flow.html** file in your server root folder.
- Open this HTML file as [http://localhost/dialog\\_flow.html](http://localhost/dialog_flow.html) and the following output will be displayed.

## Header

Welcome to Tutorialspoint!!! Click the link to throw the current page away and comes in with the next page using dialog box.

[Page Two](#)

## Footer

### Header

Click the link to go page one.

[Page One](#)

## Footer

## Page Slide Transition

---

### Description

You can slide the page from the right to the left by using the ***data-transition="slide"*** attribute.

### Example

Following example describes the use of page slide in the jQuery Mobile Framework.

```
<!DOCTYPE html>
<head>
    <meta name="viewport" content="width=device-width, initial-scale=1">
    <link rel="stylesheet"
    href="https://code.jquery.com/mobile/1.4.5/jquery.mobile-1.4.5.min.css">
    <script src="https://code.jquery.com/jquery-1.11.3.min.js"></script>
    <script src="https://code.jquery.com/mobile/1.4.5/jquery.mobile-
    1.4.5.min.js"></script>
</head>
<body>
<div data-role="page" id="page1">
    <div data-role="header">
        <h2>Header</h2>
```

```

</div>
<div data-role="main" class="ui-content">
    <p>Welcome to TutorialsPoint!!! Click the link to slide the page from right to left.</p>
    <a href="#page2" data-transition="slide">Page Two</a>
</div>
<div data-role="footer">
    <h2>Footer</h2>
</div>
</div>

<div data-role="page" id="page2">
    <div data-role="header">
        <h2>Header</h2>
    </div>

    <div data-role="main" class="ui-content">
        <p>Click the link to go page one.</p>
        <a href="#page1">Page One</a>
    </div>

    <div data-role="footer">
        <h2>Footer</h2>
    </div>
</div>
</body>
</html>

```

## Output

Let's carry out the following steps to see how the above code works:

- Save the above html code as **page\_slide.html** file in your server root folder.
- Open this HTML file as [http://localhost/page\\_slide.html](http://localhost/page_slide.html) and the following output will be displayed.

## Header

Welcome to Tutorialspoint!!! Click the link to slide the page from right to left.

[Page Two](#)

## Footer

## Header

Click the link to go page one.

[Page One](#)

## Footer

## Dialog Slide Transition

---

### Description

You can slide the page from the right to the left using the dialog box by setting the **data-transition="slide"** attribute.

### Example

Following example describes the use of dialog slide in the jQuery Mobile Framework.

```
<!DOCTYPE html>
<head>
    <meta name="viewport" content="width=device-width, initial-scale=1">
    <link rel="stylesheet"
    href="https://code.jquery.com/mobile/1.4.5/jquery.mobile-1.4.5.min.css">
    <script src="https://code.jquery.com/jquery-1.11.3.min.js"></script>
    <script src="https://code.jquery.com/mobile/1.4.5/jquery.mobile-
    1.4.5.min.js"></script>
</head>
<body>
<div data-role="page" id="page1">
    <div data-role="header">
        <h2>Header</h2>
```

```

</div>
<div data-role="main" class="ui-content">
    <p>Welcome to Tutorialspoint!!! Click the link to slide the page from right to left using dialog box.</p>
    <a href="#page2" data-transition="slide">Page Two</a>
</div>
<div data-role="footer">
    <h2>Footer</h2>
</div>
</div>

<div data-role="page" data-dialog="true" id="page2">
    <div data-role="header">
        <h2>Header</h2>
    </div>

    <div data-role="main" class="ui-content">
        <p>Click the link to go page one.</p>
        <a href="#page1">Page One</a>
    </div>

    <div data-role="footer">
        <h2>Footer</h2>
    </div>
</div>
</body>
</html>

```

## Output

Let's carry out the following steps to see how the above code works:

- Save the above html code as **dialog\_slide.html** file in your server root folder.
- Open this HTML file as [http://localhost/dialog\\_slide.html](http://localhost/dialog_slide.html) and the following output will be displayed.

## Header

Welcome to Tutorialspoint!!! Click the link to slide the page from right to left using dialog box.

[Page Two](#)

## Footer

## Header

Click the link to go page one.

[Page One](#)

## Footer

## Page Slidedefade Transition

---

### Description

You can slide the page from the right to the left and fades in the next page using the ***data-transition="slidedefade"*** attribute.

### Example

Following example describes the use of page slidedefade in the jQuery Mobile Framework.

```
<!DOCTYPE html>
<head>
    <meta name="viewport" content="width=device-width, initial-scale=1">
    <link rel="stylesheet"
    href="https://code.jquery.com/mobile/1.4.5/jquery.mobile-1.4.5.min.css">
    <script src="https://code.jquery.com/jquery-1.11.3.min.js"></script>
    <script src="https://code.jquery.com/mobile/1.4.5/jquery.mobile-
1.4.5.min.js"></script>
</head>
<body>
<div data-role="page" id="page1">
    <div data-role="header">
        <h2>Header</h2>
    </div>
```

```

<div data-role="main" class="ui-content">
    <p>Welcome to Tutorialspoint!!! Click the link to slide from right to left
    and fades in the next page.</p>
    <a href="#page2" data-transition="slidefade">Page Two</a>
</div>
<div data-role="footer">
    <h2>Footer</h2>
</div>
</div>

<div data-role="page" id="page2">
    <div data-role="header">
        <h2>Header</h2>
    </div>

    <div data-role="main" class="ui-content">
        <p>Click the link to go page one.</p>
        <a href="#page1">Page One</a>
    </div>

    <div data-role="footer">
        <h2>Footer</h2>
    </div>
</div>
</body>
</html>

```

## Output

Let's carry out the following steps to see how the above code works:

- Save the above html code as **page\_slidedefade.html** file in your server root folder.
- Open this HTML file as [http://localhost/page\\_slidedefade.html](http://localhost/page_slidedefade.html) and the following output will be displayed.

## Header

Welcome to Tutorialspoint!!! Click the link to slide from right to left and fades in the next page.

[Page Two](#)

## Footer

## Header

Click the link to go page one.

[Page One](#)

## Footer

## Dialog Slidedefade Transition

---

### Description

You can slide the page from the right to the left and fades in the next page using the dialog box by setting the ***data-transition="slidedefade"*** attribute.

### Example

Following example describes the use of dialog slidedefade in the jQuery Mobile Framework.

```
<!DOCTYPE html>
<head>
    <meta name="viewport" content="width=device-width, initial-scale=1">
    <link rel="stylesheet"
    href="https://code.jquery.com/mobile/1.4.5/jquery.mobile-1.4.5.min.css">
    <script src="https://code.jquery.com/jquery-1.11.3.min.js"></script>
    <script src="https://code.jquery.com/mobile/1.4.5/jquery.mobile-
    1.4.5.min.js"></script>
</head>
<body>
<div data-role="page" id="page1">
    <div data-role="header">
        <h2>Header</h2>
    </div>
```

```
<div data-role="main" class="ui-content">
    <p>Welcome to Tutorialspoint!!! Click the link to slide from right to left and fades in the next page using dialog box.</p>
    <a href="#page2" data-transition="slidefade">Page Two</a>
</div>
<div data-role="footer">
    <h2>Footer</h2>
</div>
</div>

<div data-role="page" data-dialog="true" id="page2">
    <div data-role="header">
        <h2>Header</h2>
    </div>

    <div data-role="main" class="ui-content">
        <p>Click the link to go page one.</p>
        <a href="#page1">Page One</a>
    </div>

    <div data-role="footer">
        <h2>Footer</h2>
    </div>
</div>
</body>
</html>
```

## Output

Let's carry out the following steps to see how the above code works:

- Save the above html code as **dialog\_slidefade.html** file in your server root folder.
- Open this HTML file as [http://localhost/dialog\\_slidefade.html](http://localhost/dialog_slidefade.html) and the following output will be displayed.

## Header

Welcome to Tutorialspoint!!! Click the link to slide from right to left and fades in the next page using dialog box.

[Page Two](#)

## Footer

## Header

Click the link to go page one.

[Page One](#)

## Footer

## Page Slideup Transition

---

### Description

You can slide the page from the bottom to up using the ***data-transition="turn"*** attribute.

### Example

Following example describes the use of page slideup in the jQuery Mobile Framework.

```
<!DOCTYPE html>
<head>
    <meta name="viewport" content="width=device-width, initial-scale=1">
    <link rel="stylesheet"
    href="https://code.jquery.com/mobile/1.4.5/jquery.mobile-1.4.5.min.css">
    <script src="https://code.jquery.com/jquery-1.11.3.min.js"></script>
    <script src="https://code.jquery.com/mobile/1.4.5/jquery.mobile-
    1.4.5.min.js"></script>
</head>
<body>
<div data-role="page" id="page1">
    <div data-role="header">
        <h2>Header</h2>
```

```
</div>
<div data-role="main" class="ui-content">
    <p>Welcome to Tutorialspoint!!! Click the link to slideup the page from right to left.</p>
    <a href="#page2" data-transition="slideup">Page Two</a>
</div>
<div data-role="footer">
    <h2>Footer</h2>
</div>
</div>

<div data-role="page" id="page2">
    <div data-role="header">
        <h2>Header</h2>
    </div>

    <div data-role="main" class="ui-content">
        <p>Click the link to go page one.</p>
        <a href="#page1">Page One</a>
    </div>

    <div data-role="footer">
        <h2>Footer</h2>
    </div>
</div>
</body>
</html>
```

## Output

Let's carry out the following steps to see how the above code works:

- Save the above html code as **page\_slideup.html** file in your server root folder.
- Open this HTML file as [http://localhost/page\\_slideup.html](http://localhost/page_slideup.html) and the following output will be displayed.

## Header

Welcome to Tutorialspoint!!! Click the link to slideup the page from right to left.

[Page Two](#)

## Footer

## Header

Click the link to go page one.

[Page One](#)

## Footer

## Dialog Slideup Transition

---

### Description

Slide the page from the bottom to up using the dialog box by setting the **data-transition="slideup"** attribute.

### Example

Following example describes the use of dialog slideup in the jQuery Mobile Framework.

```
<!DOCTYPE html>
<head>
    <meta name="viewport" content="width=device-width, initial-scale=1">
    <link rel="stylesheet"
    href="https://code.jquery.com/mobile/1.4.5/jquery.mobile-1.4.5.min.css">
    <script src="https://code.jquery.com/jquery-1.11.3.min.js"></script>
    <script src="https://code.jquery.com/mobile/1.4.5/jquery.mobile-
1.4.5.min.js"></script>
</head>
<body>
<div data-role="page" id="page1">
    <div data-role="header">
        <h2>Header</h2>
```

```

</div>
<div data-role="main" class="ui-content">
    <p>Welcome to Tutorialspoint!!! Click the link to slideup the page from
    right to left using dialog box.</p>
    <a href="#page2" data-transition="slideup">Page Two</a>
</div>
<div data-role="footer">
    <h2>Footer</h2>
</div>
</div>

<div data-role="page" data-dialog="true" id="page2">
    <div data-role="header">
        <h2>Header</h2>
    </div>

    <div data-role="main" class="ui-content">
        <p>Click the link to go page one.</p>
        <a href="#page1">Page One</a>
    </div>

    <div data-role="footer">
        <h2>Footer</h2>
    </div>
</div>
</body>
</html>

```

## Output

Let's carry out the following steps to see how the above code works:

- Save the above html code as **dialog\_slideup.html** file in your server root folder.
- Open this HTML file as [http://localhost/dialog\\_slideup.html](http://localhost/dialog_slideup.html) and the following output will be displayed.

## Header

Welcome to Tutorialspoint!!! Click the link to slideup the page from right to left using dialog box.

[Page Two](#)

## Footer

## Header

Click the link to go page one.

[Page One](#)

## Footer

## Page Slidedown Transition

---

### Description

You can slide the page from the top to bottom using the ***data-transition="slidedown"*** attribute.

### Example

Following example describes the use of page slidedown in the jQuery Mobile Framework.

```
<!DOCTYPE html>
<head>
    <meta name="viewport" content="width=device-width, initial-scale=1">
    <link rel="stylesheet"
    href="https://code.jquery.com/mobile/1.4.5/jquery.mobile-1.4.5.min.css">
    <script src="https://code.jquery.com/jquery-1.11.3.min.js"></script>
    <script src="https://code.jquery.com/mobile/1.4.5/jquery.mobile-
    1.4.5.min.js"></script>
</head>
<body>
<div data-role="page" id="page1">
    <div data-role="header">
```

```

<h2>Header</h2>
</div>
<div data-role="main" class="ui-content">
    <p>Welcome to Tutorialspoint!!! Click the link to page from top to bottom.</p>
    <a href="#page2" data-transition="slidedown">Page Two</a>
</div>
<div data-role="footer">
    <h2>Footer</h2>
</div>
</div>

<div data-role="page" id="page2">
    <div data-role="header">
        <h2>Header</h2>
    </div>

    <div data-role="main" class="ui-content">
        <p>Click the link to go page one.</p>
        <a href="#page1">Page One</a>
    </div>

    <div data-role="footer">
        <h2>Footer</h2>
    </div>
</div>
</body>
</html>

```

## Output

Let's carry out the following steps to see how the above code works:

- Save the above html code as **page\_slidedown.html** file in your server root folder.
- Open this HTML file as [http://localhost/page\\_slidedown.html](http://localhost/page_slidedown.html) and the following output will be displayed.

## Header

Welcome to Tutorialspoint!!! Click the link to page from top to bottom.

[Page Two](#)

## Footer

## Header

Click the link to go page one.

[Page One](#)

## Footer

## Dialog Slidedown Transition

---

### Description

You can slide the page from the top to bottom using the dialog box by setting **the data-transition="slidedown"** attribute.

### Example

Following example describes the use of dialog slidedown in the jQuery Mobile Framework.

```
<!DOCTYPE html>
<head>
    <meta name="viewport" content="width=device-width, initial-scale=1">
    <link rel="stylesheet"
    href="https://code.jquery.com/mobile/1.4.5/jquery.mobile-1.4.5.min.css">
    <script src="https://code.jquery.com/jquery-1.11.3.min.js"></script>
    <script src="https://code.jquery.com/mobile/1.4.5/jquery.mobile-
    1.4.5.min.js"></script>
</head>
<body>
<div data-role="page" id="page1">
    <div data-role="header">
        <h2>Header</h2>
```

```

</div>
<div data-role="main" class="ui-content">
<p>Welcome to Tutorialspoint!!! Click the link to slide page from top to bottom using dialog box.</p>
<a href="#page2" data-transition="slidedown">Page Two</a>
</div>
<div data-role="footer">
<h2>Footer</h2>
</div>
</div>

<div data-role="page" data-dialog="true" id="page2">
<div data-role="header">
<h2>Header</h2>
</div>

<div data-role="main" class="ui-content">
<p>Click the link to go page one.</p>
<a href="#page1">Page One</a>
</div>

<div data-role="footer">
<h2>Footer</h2>
</div>
</div>
</body>
</html>

```

## Output

Let's carry out the following steps to see how the above code works:

- Save the above html code as **dialog\_slidedown.html** file in your server root folder.
- Open this HTML file as [http://localhost/dialog\\_slidedown.html](http://localhost/dialog_slidedown.html) and the following output will be displayed.

## Header

Welcome to Tutorialspoint!!! Click the link to slide page from top to bottom using dialog box.

[Page Two](#)

## Footer

## Header

Click the link to go page one.

[Page One](#)

## Footer

## Page Turn Transition

---

### Description

You can turn to the next page using the ***data-transition="turn"*** attribute.

### Example

Following example describes the use of page turn in the jQuery Mobile Framework.

```
<!DOCTYPE html>
<head>
<meta name="viewport" content="width=device-width, initial-scale=1">
<link rel="stylesheet"
href="https://code.jquery.com/mobile/1.4.5/jquery.mobile-1.4.5.min.css">
<script src="https://code.jquery.com/jquery-1.11.3.min.js"></script>
<script src="https://code.jquery.com/mobile/1.4.5/jquery.mobile-
1.4.5.min.js"></script>
</head>
<body>
<div data-role="page" id="page1">
<div data-role="header">
<h2>Header</h2>
</div>
```

```

<div data-role="main" class="ui-content">
    <p>Welcome to Tutorialspoint!!! Click on the link to turn the page.</p>
    <a href="#page2" data-transition="turn">Page Two</a>
</div>

<div data-role="footer">
    <h2>Footer</h2>
</div>
</div>

<div data-role="page" id="page2">
    <div data-role="header">
        <h2>Header</h2>
    </div>

    <div data-role="main" class="ui-content">
        <p>Click the link to go page one.</p>
        <a href="#page1">Page One</a>
    </div>

    <div data-role="footer">
        <h2>Footer</h2>
    </div>
</div>
</body>
</html>

```

## Output

Let's carry out the following steps to see how the above code works:

- Save the above html code as **page\_turn.html** file in your server root folder.
- Open this HTML file as [http://localhost/page\\_turn.html](http://localhost/page_turn.html) and the following output will be displayed.

## Header

Welcome to Tutorialspoint!!! Click on the link to turn the page.

[Page Two](#)

## Footer

## Header

Click the link to go page one.

[Page One](#)

## Footer

## Dialog Turn Transition

---

### Description

You can turn to the next page using the dialog box by setting the **data-transition="turn"** attribute.

### Example

Following example describes the use of page turn *using the dialog box* in the jQuery Mobile Framework.

```
<!DOCTYPE html>
<head>
<meta name="viewport" content="width=device-width, initial-scale=1">
<link rel="stylesheet"
href="https://code.jquery.com/mobile/1.4.5/jquery.mobile-1.4.5.min.css">
<script src="https://code.jquery.com/jquery-1.11.3.min.js"></script>
<script src="https://code.jquery.com/mobile/1.4.5/jquery.mobile-
1.4.5.min.js"></script>
</head>
<body>
<div data-role="page" id="page1">
<div data-role="header">
```

```

<h2>Header</h2>
</div>
<div data-role="main" class="ui-content">
    <p>Welcome to Tutorialspoint!!! Click on the link to turn the page using dialog box.</p>
    <a href="#page2" data-transition="turn">Page Two</a>
</div>

<div data-role="footer">
    <h2>Footer</h2>
</div>
</div>

<div data-role="page" data-dialog="true" id="page2">
    <div data-role="header">
        <h2>Header</h2>
    </div>

    <div data-role="main" class="ui-content">
        <p>Click the link to go page one.</p>
        <a href="#page1">Page One</a>
    </div>

    <div data-role="footer">
        <h2>Footer</h2>
    </div>
</div>
</body>
</html>

```

## Output

Let's carry out the following steps to see how the above code works:

- Save the above html code as **dialog\_turn.html** file in your server root folder.
- Open this HTML file as [http://localhost/dialog\\_turn.html](http://localhost/dialog_turn.html) and the following output will be displayed.

## Header

Welcome to Tutorialspoint!!! Click on the link to turn the page using dialog box.

[Page Two](#)

## Footer

## Header

Click the link to go page one.

[Page One](#)

## Footer

## Page None Transition

---

### Description

You cannot use the transition effect for the page by using the ***data-transition="none"*** attribute.

### Example

Following example describes the use of page none in the jQuery Mobile Framework.

```
<!DOCTYPE html>
<head>
<meta name="viewport" content="width=device-width, initial-scale=1">
<link rel="stylesheet"
href="https://code.jquery.com/mobile/1.4.5/jquery.mobile-1.4.5.min.css">
<script src="https://code.jquery.com/jquery-1.11.3.min.js"></script>
<script src="https://code.jquery.com/mobile/1.4.5/jquery.mobile-
1.4.5.min.js"></script>
</head>
<body>
<div data-role="page" id="page1">
  <div data-role="header">
    <h2>Header</h2>
```

```

</div>
<div data-role="main" class="ui-content">
    <p>Welcome to TutorialsPoint!!! There is no transition effect after
    clicking the link.</p>
    <a href="#page2" data-transition="none">Page Two</a>
</div>

<div data-role="footer">
    <h2>Footer</h2>
</div>
</div>

<div data-role="page" id="page2">
    <div data-role="header">
        <h2>Header</h2>
    </div>

    <div data-role="main" class="ui-content">
        <p>Click the link to go page one.</p>
        <a href="#page1">Page One</a>
    </div>

    <div data-role="footer">
        <h2>Footer</h2>
    </div>
</div>
</body>
</html>

```

## Output

Let's carry out the following steps to see how the above code works:

- Save the above html code as **page\_none.html** file in your server root folder.
- Open this HTML file as [http://localhost/page\\_none.html](http://localhost/page_none.html) and the following output will be displayed.

## Header

Welcome to Tutorialspoint!!! There is no transition effect after clicking the link.

[Page Two](#)

## Footer

## Header

Click the link to go page one.

[Page One](#)

## Footer

## Dialog None Transition

---

### Description

You cannot use transition effect for the dialog box by setting the **data-transition="none"** attribute.

### Example

Following example describes the use of dialog none in the jQuery Mobile Framework.

```
<!DOCTYPE html>
<head>
<meta name="viewport" content="width=device-width, initial-scale=1">
<link rel="stylesheet"
href="https://code.jquery.com/mobile/1.4.5/jquery.mobile-1.4.5.min.css">
<script src="https://code.jquery.com/jquery-1.11.3.min.js"></script>
<script src="https://code.jquery.com/mobile/1.4.5/jquery.mobile-
1.4.5.min.js"></script>
</head>
<body>
<div data-role="page" id="page1">
<div data-role="header">
<h2>Header</h2>
</div>
```

```

<div data-role="main" class="ui-content">
    <p>Welcome to Tutorialspoint!!! Displays the dialog box with no transition effect after clicking the link.</p>
    <a href="#page2" data-transition="none">Page Two</a>
</div>

<div data-role="footer">
    <h2>Footer</h2>
</div>
</div>

<div data-role="page" data-dialog="true" id="page2">
    <div data-role="header">
        <h2>Header</h2>
    </div>

    <div data-role="main" class="ui-content">
        <p>Click the link to go page one.</p>
        <a href="#page1">Page One</a>
    </div>

    <div data-role="footer">
        <h2>Footer</h2>
    </div>
</div>
</body>
</html>

```

## Output

Let's carry out the following steps to see how the above code works:

- Save the above html code as **dialog\_none.html** file in your server root folder.
- Open this HTML file as [http://localhost/dialog\\_none.html](http://localhost/dialog_none.html) and the following output will be displayed.

## Header

Welcome to Tutorialspoint!!! Displays the dialog box with no transition effect after clicking the link.

[Page Two](#)

## Footer

## Header

Click the link to go page one.

[Page One](#)

## Footer

## Setting Transition and Global Configuration

By default, pages will have **fade** transition in the framework. You can use custom transitions by adding the **data-transition** attribute to the link. You can use different default transition effects for the page using **defaultPageTransition** option globally. For dialogs, you can make use of **defaultDialogTransition** option.

## Fallback Transition

All transitions support 3D transformations except the fade transition. The devices which do not support 3D transformation, they will have to make use of fade transition. Some browsers do not support 3D transformations for each transition type, so you can use the **fade** as default transition fallback.

## Max Scroll for Transitions

The transitions are set to **none** when you are scrolling from or to a page and the scroll position will be three times the height of the device screen. Sometimes, you may get slow in responding or the browser may crash when you click any navigation element; so to avoid this we are using the scroll position for transition by using **getMaxScrollForTransition** function.

## Max Width for Transitions

You can disable the transition when the window width is higher than the pixel width. You can configure the max width for transitions using the **`$.mobile.maxTransitionWidth`** global option, which is set to false by default. It takes values such as pixel width or false value, and the transition will be set to none if it is not a false value when the window is higher than the specified value.

## Same Page Transition

You can add the transitions to the current page using the **`allowSamePageTransition`** option of page container widget's **`change()`** method.

## Creating Custom Transitions

You can create the custom transitions in the page using the **`$.mobile.transitionHandlers`** option that expands the selection of transitions on the website or application.

## 6. jQuery Mobile – Layouts

Grid systems are used to create page layouts through a series of rows and columns that house your content.

Following table demonstrates the types of grids in detail.

| Sr. No. | Types & Description   |
|---------|---|
| 1       | <b><u>Grid</u></b><br>jQuery mobile grid system creates page layouts through a series of rows and columns.                                      |
| 2       | <b><u>Buttons in grids</u></b><br>Collection of button in grid format in jQuery mobile.   |
| 3       | <b><u>Custom responsive grid</u></b><br>The basic grid style can be easily extended to the custom responsive layout using media queries in CSS. |

### Grid Layouts

---

#### Description

jQuery mobile grid system creates page layouts through a series of rows and columns.

Following table demonstrates the types of grid in detail.

| Sr. No. | Types & Description   |
|---------|---|
| 1       | <b><u>Basic grids</u></b><br>Include <i>ui-grid-solo</i> class in <div> to create a single column grid, the grid width is 100%.   |
| 2       | <b><u>Two-column grids</u></b><br>Add class <i>ui-grid-a</i> to a <div> and include two child container with class <i>ui-block-a</i> and <i>ui-block-b</i> to create two column layout. |
| 3       | <b><u>Three-column grids</u></b><br>The <i>ui-grid-b</i> class is used to create a three column grid, it includes three child container with class <i>ui-block-a/b/c</i> .              |

|   |   |
|---|---|
| 4 | <b><u>Four-column grids</u></b><br>The <i>ui-grid-c</i> class is used to create a four column grid, it includes four child container with class <i>ui-block-a/b/c/d/e</i> . |
| 5 | <b><u>Five-column grids</u></b><br>The <i>ui-grid-d</i> class is used to create a five column grid, it includes five child container with class <i>ui-block-a/b/c/d</i> .   |
| 6 | <b><u>Multiple row grids</u></b><br>Multiple rows can be wrapped in the grid by specifying any column class on a container.   |
| 7 | <b><u>Grid solo class</u></b><br>The standard grid is made responsive by overriding the grid styles below a single breakpoint.  |
| 8 | <b><u>Responsive grids</u></b><br>The stack grids below 35em (560px), a preset breakpoint can be applied.   |
| 9 | <b><u>Breakpoint preset</u></b><br>The stack grids below 35em (560px), a preset breakpoint can be applied. In grid container, include <i>.ui-responsive</i> class.          |

## Basic Grid Layouts

---

### Description

Include ***ui-grid-solo*** class in <div> to create a single column grid, the grid width is 100%.

### Example

Following example demonstrates the use of basic grid in the jQuery Mobile.

```
<!DOCTYPE html>
<html>
<head>
    <meta name="viewport" content="width=device-width, initial-scale=1">
    <link rel="stylesheet"
    href="https://code.jquery.com/mobile/1.4.5/jquery.mobile-1.4.5.min.css">
    <script src="https://code.jquery.com/jquery-1.11.3.min.js"></script>
    <script src="https://code.jquery.com/mobile/1.4.5/jquery.mobile-
1.4.5.min.js"></script>
</head>
<body>
```

```

<h2>Basic Grid Example</h2>
<div class="ui-grid-solo">
  <div class="ui-block-a"><div class="ui-bar ui-bar-a">First Block</div></div>
</div>

<h2>Basic Grid button Example</h2>
<fieldset class="ui-grid-solo">
  <div class="ui-block-a"><a class="ui-btn ui-corner-all ui-shadow ui-btn-b">Button 1</a></div>
</fieldset>
</body>
</html>

```

## Output

Let's carry out the following steps to see how the above code works:

- Save the above html code as **basic\_grid.html** file in your server root folder.
- Open this HTML file as [http://localhost/basic\\_grid.html](http://localhost/basic_grid.html) and the following output will be displayed.

## Basic Grid Example

First Block

## Basic Grid button Example

Button 1

## Two-column Grids

---

### Description

Add class **ui-grid-a** to a <div> and include two child container with class **ui-block-a** and **ui-block-b** to create two column layout. Both grid width will be 50% each. The **ui-bar-a** is used to apply the background and font styling.

### Example

Following example demonstrates the use of two column grid in the jQuery Mobile.

```
<!DOCTYPE html>
<html>
<head>
    <meta name="viewport" content="width=device-width, initial-scale=1">
    <link rel="stylesheet"
    href="https://code.jquery.com/mobile/1.4.5/jquery.mobile-1.4.5.min.css">
    <script src="https://code.jquery.com/jquery-1.11.3.min.js"></script>
    <script src="https://code.jquery.com/mobile/1.4.5/jquery.mobile-
1.4.5.min.js"></script>
</head>
<body>
<h2>Two Column Grid Example</h2>
<div class="ui-grid-a">
    <div class="ui-block-a"><div class="ui-bar ui-bar-a">First Block</div></div>
    <div class="ui-block-b"><div class="ui-bar ui-bar-b">Second
    Block</div></div>
</div>

<h2>Two Column Grid button Example</h2>
<fieldset class="ui-grid-a">
    <div class="ui-block-a"><a class="ui-btn ui-corner-all ui-shadow ui-btn-
a">Button 1</a></div>
    <div class="ui-block-b"><a class="ui-btn ui-corner-all ui-shadow ui-btn-
b">Button 2</a></div>
</fieldset>
</body>
</html>
```

## Output

Let's carry out the following steps to see how the above code works:

- Save the above html code as **two\_column\_grid.html** file in your server root folder.
- Open this HTML file as [http://localhost/two\\_column\\_grid.html](http://localhost/two_column_grid.html) and the following output will be displayed.

## Two Column Grid Example

First Block  
Second Block

## Two Column Grid button Example

Button 1  
Button 2

## Three-column Grids

---

### Description

The ***ui-grid-b*** class is used to create a three-column grid. It includes three child container with class ***ui-block-a/b/c***. All grid column width will be 33% each.

### Example

Following example demonstrates the use of three column grid in the jQuery Mobile.

```
<!DOCTYPE html>
<html>
<head>
    <meta name="viewport" content="width=device-width, initial-scale=1">
    <link rel="stylesheet"
    href="https://code.jquery.com/mobile/1.4.5/jquery.mobile-1.4.5.min.css">
    <script src="https://code.jquery.com/jquery-1.11.3.min.js"></script>
    <script src="https://code.jquery.com/mobile/1.4.5/jquery.mobile-
    1.4.5.min.js"></script>
</head>
<body>
<h2>Three Column Grid Example</h2>
<div class="ui-grid-b">
    <div class="ui-block-a"><div class="ui-bar ui-bar-a">First Block</div></div>
    <div class="ui-block-b"><div class="ui-bar ui-bar-b">Second
    Block</div></div>
    <div class="ui-block-c"><div class="ui-bar ui-bar-a">Third Block</div></div>
</div>
```

```
<h2>Three Column Grid button Example</h2>
<fieldset class="ui-grid-b">
  <div class="ui-block-a"><a class="ui-btn ui-corner-all ui-shadow ui-btn-b">Button 1</a></div>
  <div class="ui-block-b"><a class="ui-btn ui-corner-all ui-shadow ui-btn-a">Button 2</a></div>
  <div class="ui-block-c"><a class="ui-btn ui-corner-all ui-shadow ui-btn-b">Button 3</a></div>
</fieldset>
</body>
</html>
```

## Output

Let's carry out the following steps to see how the above code works:

- Save the above html code as **three\_column\_grid.html** file in your server root folder.
- Open this HTML file as [http://localhost/three\\_column\\_grid.html](http://localhost/three_column_grid.html) and the following output will be displayed.

## Four-column Grids

---

### Description

The **ui-grid-c** class is used to create a four-column grid. It includes four child container with class **ui-block-a/b/c/d/e**. All grid column width will be 25% each.

### Example

Following example demonstrates the use of four column grid in the jQuery Mobile.

```
<!DOCTYPE html>
<html>
<head>
  <meta name="viewport" content="width=device-width, initial-scale=1">
  <link rel="stylesheet"
  href="https://code.jquery.com/mobile/1.4.5/jquery.mobile-1.4.5.min.css">
  <script src="https://code.jquery.com/jquery-1.11.3.min.js"></script>
  <script src="https://code.jquery.com/mobile/1.4.5/jquery.mobile-
  1.4.5.min.js"></script>
</head>
<body>
```

```

<h2>Four Column Grid Example</h2>
<div class="ui-grid-c">
  <div class="ui-block-a"><div class="ui-bar ui-bar-a">First Block</div></div>
  <div class="ui-block-b"><div class="ui-bar ui-bar-b">Second
  Block</div></div>
  <div class="ui-block-c"><div class="ui-bar ui-bar-a">Third Block</div></div>
  <div class="ui-block-d"><div class="ui-bar ui-bar-b">Fourth
  Block</div></div>
</div>

<h2>Four Column Grid button Example</h2>
<fieldset class="ui-grid-c">
  <div class="ui-block-a"><a class="ui-btn ui-corner-all ui-shadow ui-btn-
b">Button 1</a></div>
  <div class="ui-block-b"><a class="ui-btn ui-corner-all ui-shadow ui-btn-
a">Button 2</a></div>
  <div class="ui-block-c"><a class="ui-btn ui-corner-all ui-shadow ui-btn-
b">Button 3</a></div>
  <div class="ui-block-d"><a class="ui-btn ui-corner-all ui-shadow ui-btn-
a">Button 4</a></div>
</fieldset>
</body>
</html>

```

## Output

Let's carry out the following steps to see how the above code works:

- Save the above html code as **four\_column\_grid.html** file in your server root folder.
- Open this HTML file as [http://localhost/four\\_column\\_grid.html](http://localhost/four_column_grid.html) and the following output will be displayed.

## Four Column Grid Example

First Block  
Second Block  
Third Block  
Fourth Block

## Four Column Grid button Example

Button 1  
Button 2  
Button 3  
Button 4

## Five-column Grids

---

### Description

The ***ui-grid-d*** class is used to create a five-column grid. It includes five child container with class ***ui-block-a/b/c/d***. All grid column width will be 20% each.

### Example

Following example demonstrates the use of five column grid in the jQuery Mobile.

```
<!DOCTYPE html>
<html>
<head>
    <meta name="viewport" content="width=device-width, initial-scale=1">
    <link rel="stylesheet"
    href="https://code.jquery.com/mobile/1.4.5/jquery.mobile-1.4.5.min.css">
    <script src="https://code.jquery.com/jquery-1.11.3.min.js"></script>
    <script src="https://code.jquery.com/mobile/1.4.5/jquery.mobile-
    1.4.5.min.js"></script>
</head>
<body>
    <h2>Five Column Grid Example</h2>
    <div class="ui-grid-d">
        <div class="ui-block-a"><div class="ui-bar ui-bar-a">First Block</div></div>
        <div class="ui-block-b"><div class="ui-bar ui-bar-b">Second
        Block</div></div>
```

```

<div class="ui-block-c"><div class="ui-bar ui-bar-a">Third Block</div></div>

<div class="ui-block-d"><div class="ui-bar ui-bar-b">Fourth
Block</div></div>

<div class="ui-block-e"><div class="ui-bar ui-bar-a">Fourth
Block</div></div>

</div>

<h2>Five Column Grid button Example</h2>
<fieldset class="ui-grid-d">
  <div class="ui-block-a"><a class="ui-btn ui-corner-all ui-shadow ui-btn-
b">Button 1</a></div>
  <div class="ui-block-b"><a class="ui-btn ui-corner-all ui-shadow ui-btn-
a">Button 2</a></div>
  <div class="ui-block-c"><a class="ui-btn ui-corner-all ui-shadow ui-btn-
b">Button 3</a></div>
  <div class="ui-block-d"><a class="ui-btn ui-corner-all ui-shadow ui-btn-
a">Button 4</a></div>
  <div class="ui-block-e"><a class="ui-btn ui-corner-all ui-shadow ui-btn-
b">Button 5</a></div>
</fieldset>
</body>
</html>

```

## Output

Let's carry out the following steps to see how the above code works:

- Save the above html code as **five\_column\_grid.html** file in your server root folder.
- Open this HTML file as [http://localhost/five\\_column\\_grid.html](http://localhost/five_column_grid.html) and the following output will be displayed.

## Five Column Grid Example

First Block  
 Second Block  
 Third Block  
 Fourth Block  
 Fourth Block

## Five Column Grid button Example

Button 1  
 Button 2  
 Button 3  
 Button 4  
 Button 5

## Multiple Row Grid

### Description

Multiple rows can be wrapped in the grid by specifying any column class on a container, i.e. ***ui-grid-a***, ***ui-grid-b***, ***ui-grid-c***, ***ui-grid-d*** or ***ui-grid-b***. Include the child container inside it with class ***ui-block-a/b/c/d*** repeatedly for the number of rows you want in it.

### Example

Following example demonstrates the use of multiple row grid in the jQuery Mobile.

```
<!DOCTYPE html>
<html>
<head>
    <meta name="viewport" content="width=device-width, initial-scale=1">
    <link rel="stylesheet"
    href="https://code.jquery.com/mobile/1.4.5/jquery.mobile-1.4.5.min.css">
    <script src="https://code.jquery.com/jquery-1.11.3.min.js"></script>
    <script src="https://code.jquery.com/mobile/1.4.5/jquery.mobile-
    1.4.5.min.js"></script>
</head>
<body>
<h2>Multiple Row Grid Example</h2>
<div class="ui-grid-b">
```

```

<div class="ui-block-a"><div class="ui-bar ui-bar-b">First Block</div></div>
<div class="ui-block-b"><div class="ui-bar ui-bar-a">Second
Block</div></div>
<div class="ui-block-c"><div class="ui-bar ui-bar-b">Third Block</div></div>
<div class="ui-block-a"><div class="ui-bar ui-bar-a">First Block</div></div>
<div class="ui-block-b"><div class="ui-bar ui-bar-b">Second
Block</div></div>
<div class="ui-block-c"><div class="ui-bar ui-bar-a">Third Block</div></div>
<div class="ui-block-a"><div class="ui-bar ui-bar-b">First Block</div></div>
<div class="ui-block-b"><div class="ui-bar ui-bar-a">Second
Block</div></div>
<div class="ui-block-c"><div class="ui-bar ui-bar-b">Third Block</div></div>
</div>
</body>
</html>

```

## Output

Let's carry out the following steps to see how the above code works:

- Save the above html code as **multiple\_row\_grid.html** file in your server root folder.
- Open this HTML file as [http://localhost/multiple\\_row\\_grid.html](http://localhost/multiple_row_grid.html) and the following output will be displayed.

## Multiple Row Grid Example

First Block  
 Second Block  
 Third Block  
 First Block  
 Second Block  
 Third Block  
 First Block  
 Second Block  
 Third Block  
 First Block  
 Second Block  
 Third Block

## Grid Solo Class

### Description

**ui-grid-solo** class is used to create a single column with class **ui-block-a** as a child container.

### Example

Following example demonstrates the use of grid solo class in the jQuery Mobile.

```
<!DOCTYPE html>
<html>
<head>
    <meta name="viewport" content="width=device-width, initial-scale=1">
    <link rel="stylesheet"
    href="https://code.jquery.com/mobile/1.4.5/jquery.mobile-1.4.5.min.css">
    <script src="https://code.jquery.com/jquery-1.11.3.min.js"></script>
    <script src="https://code.jquery.com/mobile/1.4.5/jquery.mobile-
    1.4.5.min.js"></script>
</head>
<body>
<h2>Grid Solo Class Example</h2>
<div class="ui-grid-solo">
    <div class="ui-block-a"><a class="ui-btn ui-corner-all ui-shadow ui-btn-
    a">Button 1</a></div>
```

```

<div class="ui-block-b"><a class="ui-btn ui-corner-all ui-shadow ui-btn-b">Button 2</a></div>

</div>
<div class="ui-grid-solo">
    <div class="ui-block-a"><input type="button" value="More"></div>
</div>
</body>
</html>
```

## Output

Let's carry out the following steps to see how the above code works:

- Save the above html code as **grid\_solo\_class.html** file in your server root folder.
- Open this HTML file as [http://localhost/grid\\_solo\\_class.html](http://localhost/grid_solo_class.html) and the following output will be displayed.

## Grid Solo Class Example

Button 1

Button 2

**More**

## Responsive Grids

The standard grid is made responsive by overriding the grid styles below a single breakpoint. **max-width** is used as an override to apply the stacked styling. To scope the styles for media query include **class my-breakpoint**. You can include additional breakpoint or customize the appearance further.

```
@media all and (max-width: 35em) {
    .my-breakpoint .ui-block-a,
    .my-breakpoint .ui-block-b,
    .my-breakpoint .ui-block-c,
    .my-breakpoint .ui-block-d,
    .my-breakpoint .ui-block-e {
        width: 100%;
        float: none;
    }
}
```

## Breakpoint Preset

---

### Description

To stack grids below 35em (560px), a preset breakpoint can be applied. In grid container include **.ui-responsive** class.

### Example

Following example demonstrates the use of responsive class in the jQuery Mobile.

```
<!DOCTYPE html>
<html>
<head>
    <meta name="viewport" content="width=device-width, initial-scale=1">
    <link rel="stylesheet"
    href="https://code.jquery.com/mobile/1.4.5/jquery.mobile-1.4.5.min.css">
    <script src="https://code.jquery.com/jquery-1.11.3.min.js"></script>
    <script src="https://code.jquery.com/mobile/1.4.5/jquery.mobile-
    1.4.5.min.js"></script>
</head>
<body>
<h2>Two Column Responsive Example</h2>
<div class="ui-grid-a ui-responsive">
    <div class="ui-block-a"><div class="ui-body ui-body-a">First
    Block</div></div>
    <div class="ui-block-b"><div class="ui-body ui-body-b">Second
    Block</div></div>
</div>
```

```

<h2>Three Column Responsive Example</h2>
<div class="ui-grid-b ui-responsive">
  <div class="ui-block-a"><div class="ui-body ui-body-a">First
  Block</div></div>
  <div class="ui-block-b"><div class="ui-body ui-body-b">Second
  Block</div></div>
  <div class="ui-block-c"><div class="ui-body ui-body-a">Third
  Block</div></div>
</div>

<h2>Four Column Responsive Example</h2>
<div class="ui-grid-c ui-responsive">
  <div class="ui-block-a"><div class="ui-body ui-body-a">First
  Block</div></div>
  <div class="ui-block-b"><div class="ui-body ui-body-b">Second
  Block</div></div>
  <div class="ui-block-c"><div class="ui-body ui-body-a">Third
  Block</div></div>
  <div class="ui-block-d"><div class="ui-body ui-body-b">Fourth
  Block</div></div>
</div>

<h2>Five Column Responsive Example</h2>
<div class="ui-grid-d ui-responsive">
  <div class="ui-block-a"><div class="ui-body ui-body-a">First
  Block</div></div>
  <div class="ui-block-b"><div class="ui-body ui-body-b">Second
  Block</div></div>
  <div class="ui-block-c"><div class="ui-body ui-body-a">Third
  Block</div></div>
  <div class="ui-block-d"><div class="ui-body ui-body-b">Fourth
  Block</div></div>
  <div class="ui-block-e"><div class="ui-body ui-body-a">Fifth
  Block</div></div>
</div>

</body>
</html>

```

## Output

Let's carry out the following steps to see how the above code works:

- Save the above html code as **breakpoint\_preset.html** file in your server root folder.
- Open this HTML file as [http://localhost/breakpoint\\_preset.html](http://localhost/breakpoint_preset.html) and the following output will be displayed.

## Two Column Responsive Example

First Block  
Second Block

## Three Column Responsive Example

First Block  
Second Block  
Third Block

## Four Column Responsive Example

First Block  
Second Block  
Third Block  
Fourth Block

## Five Column Responsive Example

First Block  
Second Block  
Third Block  
Fourth Block  
Fifth Block

## Buttons in Grids

---

### Description

It is a collection of buttons in grid format in jQuery mobile.

Following table demonstrates the types of buttons in grids in detail.

| Sr. No. | Types & Description  |
|---------|--|
| 1       | <b><u>Basic</u></b><br>Margin gap is there on the left and right side of the button.   |
| 2       | <b><u>Mini sized</u></b><br>In the element, include attribute <i>data-mini="true"</i> for creating a mini version button.            |
| 3       | <b><u>Icon only, inline centered</u></b><br>The only icon is set on the button by using the class <i>ui-icon-*</i> in jQuery mobile. |
| 4       | <b><u>Responsive</u></b><br>To stack grids below 35em (560px), a preset breakpoint can be applied.                                   |
| 5       | <b><u>Alignment</u></b><br>A single button is aligned along with the buttons present in other grids.                                 |

## Basic Buttons in Grids

---

### Description

Margin gap is there on the left and right side of the button. Margin is not applied directly to the elements when the width is 100%, wrap the button inside the ***div*** tag to get the margin space as of the other button.

### Example

Following example demonstrates the use of basic button grid in the jQuery Mobile.

```
<!DOCTYPE html>
<html>
<head>
    <meta name="viewport" content="width=device-width, initial-scale=1">
    <link rel="stylesheet"
    href="https://code.jquery.com/mobile/1.4.5/jquery.mobile-1.4.5.min.css">
    <script src="https://code.jquery.com/jquery-1.11.3.min.js"></script>
    <script src="https://code.jquery.com/mobile/1.4.5/jquery.mobile-
1.4.5.min.js"></script>
</head>
<body>
<h2>Basic Button Grid Example</h2>
```

```

<div class="ui-grid-b">
    <div class="ui-block-a"><a class="ui-btn ui-corner-all ui-shadow ui-btn-b">Using Anchor Tag</a></div>
    <div class="ui-block-b"><button class="ui-btn ui-corner-all ui-shadow ui-btn-a">Using Button</button></div>
    <div class="ui-block-c"><input type="button" value="Using Input Tag"></div>
</div>

<div class="ui-grid-b">
    <div class="ui-block-a">
        <label for="grid-select-1" class="ui-hidden-accessible">Select</label>
        <select id="grid-select-1">
            <option>Select</option>
            <option value="1">First Option</option>
            <option value="2">Second Option</option>
            <option value="3">Third Option</option>
        </select>
    </div>
    <div class="ui-block-b">
        <label for="grid-checkbox-1">Checkbox</label>
        <input type="checkbox" id="grid-checkbox-1">
    </div>
    <div class="ui-block-c">
        <label for="grid-radio-1">Radio</label>
        <input type="radio" id="grid-radio-1">
    </div>
</div>
</body>
</html>

```

## Output

Let's carry out the following steps to see how the above code works:

- Save the above html code as **button\_grid\_basic.html** file in your server root folder.

- Open this HTML file as `http://localhost/button_grid_basic.html` and the following output is displayed.

## Basic Button Grid Example

Using Anchor Tag

Using Button

Using Input Tag

Select | Select ▾

Checkbox

Radio

## Mini-sized Buttons in Grids

### Description

In an element, include the attribute `data-mini="true"` for creating a mini version of the button. It is useful in toolbars and tight spaces for more compact version.

### Example

Following example demonstrates the use of mini-sized button grid in the jQuery Mobile.

```
<!DOCTYPE html>
<html>
<head>
  <meta name="viewport" content="width=device-width, initial-scale=1">
  <link rel="stylesheet"
  href="https://code.jquery.com/mobile/1.4.5/jquery.mobile-1.4.5.min.css">
  <script src="https://code.jquery.com/jquery-1.11.3.min.js"></script>
  <script src="https://code.jquery.com/mobile/1.4.5/jquery.mobile-
1.4.5.min.js"></script>
</head>
<body>
<h2>Mini Size Button Grid Example</h2>
<div class="ui-grid-b">
  <div class="ui-block-a"><a class="ui-btn ui-corner-all ui-shadow ui-btn-b
ui-mini">Using Anchor Tag</a></div>
```

```

<div class="ui-block-b"><button class="ui-btn ui-corner-all ui-shadow ui-btn-a ui-mini">Using Button</button></div>

<div class="ui-block-c"><input type="button" value="Using Input Tag" data-mini="true"></div>

</div>

<div class="ui-grid-b">
  <div class="ui-block-a">
    <label for="grid-select-1" class="ui-hidden-accessible">Select</label>
    <select id="grid-select-1" data-mini="true">
      <option>Select</option>
      <option value="1">First Option</option>
      <option value="2">Second Option</option>
      <option value="3">Third Option</option>
    </select>
  </div>
  <div class="ui-block-b">
    <label for="grid-checkbox-1">Checkbox</label>
    <input type="checkbox" id="grid-checkbox-1" data-mini="true">
  </div>
  <div class="ui-block-c">
    <label for="grid-radio-1">Radio</label>
    <input type="radio" id="grid-radio-1" data-mini="true">
  </div>
</div>
</body>
</html>

```

## Output

Let's carry out the following steps to see how the above code works:

- Save the above html code as **button\_grid\_mini\_sized.html** file in your server root folder.

- Open this HTML file as [http://localhost/button\\_grid\\_mini\\_sized.html](http://localhost/button_grid_mini_sized.html) and the following output will be displayed.

## Mini Size Button Grid Example

Using Anchor Tag

[Using Button](#)

[Using Input Tag](#)

Select  Select ▾

Checkbox

Radio

## Icon Only, Inline Centered Buttons in Grids

### Description

An icon can be set on the button by using the **class ui-icon-\***. Class **ui-btn-inline** is used to display the button inline. This means that it will only consume as much space as is needed for the label.

### Example

Following example demonstrates the use of icon button grid in the jQuery Mobile.

```
<!DOCTYPE html>
<html>
<head>
    <meta name="viewport" content="width=device-width, initial-scale=1">
    <link rel="stylesheet"
    href="https://code.jquery.com/mobile/1.4.5/jquery.mobile-1.4.5.min.css">
    <script src="https://code.jquery.com/jquery-1.11.3.min.js"></script>
    <script src="https://code.jquery.com/mobile/1.4.5/jquery.mobile-
1.4.5.min.js"></script>
</head>
<body>
<h2>Icons Only Inline Center Button Grid Example</h2>
<div class="ui-grid-c">
    <div class="ui-block-a"><a class="ui-btn ui-corner-all ui-icon-arrow-l ui-
btn-icon-notext ui-btn-inline">Button</a></div>
    <div class="ui-block-b"><a class="ui-btn ui-corner-all ui-icon-home ui-btn-
icon-notext ui-btn-inline">Button</a></div>

```

```

<div class="ui-block-b"><a class="ui-btn ui-corner-all ui-icon-gear ui-btn-icon-notext ui-btn-inline">Button</a></div>
<div class="ui-block-e"><a class="ui-btn ui-corner-all ui-icon-arrow-r ui-btn-icon-notext ui-btn-inline">Button</a></div>
</div>
</body>
</html>

```

## Output

Let's carry out the following steps to see how the above code works:

- Save the above html code as **button\_grid\_inline\_centered.html** file in your server root folder.
- Open this HTML file as [http://localhost/button\\_grid\\_inline\\_centered.html](http://localhost/button_grid_inline_centered.html) and the following output will be displayed.

## Icons Only Inline Center Button Grid Example

```

Button
Button
Button
Button

```

## Responsive Buttons in Grids

### Description

To stack grids below 35em (560px), a preset breakpoint can be applied. In button grid container include **.ui-responsive** class.

### Example

Following example demonstrates the use of button grid responsive in the jQuery Mobile.

```

<!DOCTYPE html>
<html>
<head>
<meta name="viewport" content="width=device-width, initial-scale=1">

```

```

<link rel="stylesheet"
href="https://code.jquery.com/mobile/1.4.5/jquery.mobile-1.4.5.min.css">
<script src="https://code.jquery.com/jquery-1.11.3.min.js"></script>
<script src="https://code.jquery.com/mobile/1.4.5/jquery.mobile-
1.4.5.min.js"></script>
</head>
<body>
<h2>Responsive grid Example Using Anchor Tag</h2>
<div class="ui-grid-b ui-responsive">
    <div class="ui-block-a"><a class="ui-btn ui-corner-all ui-btn-icon-left ui-
icon-arrow-l">Previous</a></div>
    <div class="ui-block-b"><a class="ui-btn ui-corner-all ui-icon-plus ui-btn-
icon-left">Add </a></div>
    <div class="ui-block-c"><a class="ui-btn ui-corner-all ui-btn-icon-right ui-
icon-arrow-r">Next</a></div>
</div>

<h2>Responsive Grid Example Using Input</h2>
<div class="ui-grid-b ui-responsive">
    <div class="ui-block-a"><input type="button" data-icon="plus"
value="Add"></div>
    <div class="ui-block-b"><input type="button" data-icon="check"
value="Confirm"></div>
    <div class="ui-block-c"><input type="button" data-icon="minus"
value="Minus"></div>
</div>
</body>
</html>

```

## Output

Let's carry out the following steps to see how the above code works:

- Save the above html code as **button\_grid\_responsive.html** file in your server root folder.
- Open this HTML file as [http://localhost/button\\_grid\\_responsive.html](http://localhost/button_grid_responsive.html) and the following output will be displayed.

## Responsive grid Example Using Anchor Tag

[Previous](#)

[Add](#)

[Next](#)

## Responsive Grid Example Using Input

Add  
Confirm  
Minus

### Alignment Buttons in Grids

#### Description

A single button is aligned along with the buttons present in other grids.

#### Example

Following example demonstrates the use of button grid alignment in the jQuery Mobile.

```
<!DOCTYPE html>
<html>
<head>
    <meta name="viewport" content="width=device-width, initial-scale=1">
    <link rel="stylesheet"
    href="https://code.jquery.com/mobile/1.4.5/jquery.mobile-1.4.5.min.css">
    <script src="https://code.jquery.com/jquery-1.11.3.min.js"></script>
    <script src="https://code.jquery.com/mobile/1.4.5/jquery.mobile-
    1.4.5.min.js"></script>
</head>
<body>
<h2>Responsive grid Example Using Anchor Tag</h2>
<div class="ui-grid-b ui-responsive">
    <div class="ui-block-a"><a class="ui-btn ui-corner-all">Block A</a></div>
    <div class="ui-block-b"><a class="ui-btn ui-corner-all">Block B </a></div>
    <div class="ui-block-c"><a class="ui-btn ui-corner-all">Block C</a></div>
</div>
```

```

<div class="ui-grid-a ui-responsive">
  <div class="ui-block-a"><a class="ui-btn ui-corner-all">Block A</a></div>
  <div class="ui-block-b"><a class="ui-btn ui-corner-all">Block B </a></div>
</div>

<div class="ui-grid-solo ui-responsive">
  <div class="ui-block-a"><a class="ui-btn ui-corner-all">Block A</a></div>
</div>

<a class="ui-btn ui-corner-all">Out of div tag</a>
</body>
</html>

```

## Output

Let's carry out the following steps to see how the above code works:

- Save the above html code as **button\_grid\_alignment.html** file in your server root folder.
- Open this HTML file as [http://localhost/button\\_grid\\_alignment.html](http://localhost/button_grid_alignment.html) and the following output will be displayed.

## Responsive grid Example Using Anchor Tag

Block A  
 Block B  
 Block C  
 Block A  
 Block B  
 Block A  
 Out of div tag

## Custom Responsive Grid

---

### Description

The basic grid style can be easily extended to the custom responsive layout using media queries in CSS.

## Example

Following example demonstrates the use of custom responsive grid in the jQuery Mobile.

```
<!DOCTYPE html>
<html>
<head>
    <meta name="viewport" content="width=device-width, initial-scale=1">
    <link rel="stylesheet"
    href="https://code.jquery.com/mobile/1.4.5/jquery.mobile-1.4.5.min.css">
    <script src="https://code.jquery.com/jquery-1.11.3.min.js"></script>
    <script src="https://code.jquery.com/mobile/1.4.5/jquery.mobile-
1.4.5.min.js"></script>
<style>
    .rwd-example .ui-block-b, .rwd-example .ui-block-c {
        float: left;
        width: 25%;
    }

    .rwd-example .ui-block-a{
        float: left;
        width: 49.95%;
    }
</style>
</head>
<body>
<h2>Custom Responsive Grid</h2>
<div class="rwd-example">
    <div class="ui-block-a">
        <div class="ui-body ui-body-a">

            <h3>First Block</h3>
            <p>Lorem Ipsum is simply dummy text of the printing and typesetting industry. Lorem Ipsum has been the industry's standard dummy text ever since the 1500s, when an unknown printer took a galley of type and scrambled it to make a type specimen book. It has survived not only five centuries, but also

```

```

the leap into electronic typesetting, remaining essentially unchanged. It was
popularised in the 1960s with the release of Letraset sheets containing Lorem
Ipsum passages, and more recently with desktop publishing software like
Aldus</p>
</div>
</div>
<div class="ui-block-b">
<div class="ui-body ui-body-a">
<h3>Second Block</h3>
<p>Lorem Ipsum is simply dummy text of the printing and typesetting
industry. Lorem Ipsum has been the industry's standard dummy text ever since
the 1500s, when an unknown printer took a galley of type and scrambled it to
make a type specimen book.</p>
</div>
</div>
<div class="ui-block-c">
<div class="ui-body ui-body-a">
<h3>Third Block</h3>
<p>Lorem Ipsum is simply dummy text of the printing and typesetting
industry. Lorem Ipsum has been the industry's standard dummy text ever since
the 1500s, when an unknown printer took a galley of type and scrambled it to
make a type specimen book.</p>
</div>
</div>
</div>
</body>
</html>
```

## Output

Let's carry out the following steps to see how the above code works:

- Save the above html code as **custom\_responsive\_grid.html** file in your server root folder.

- Open this HTML file as [http://localhost/custom\\_responsive\\_grid.html](http://localhost/custom_responsive_grid.html) and the following output will be displayed.

## Custom Responsive Grid

### First Block

Lorem Ipsum is simply dummy text of the printing and typesetting industry. Lorem Ipsum has been the industry's standard dummy text ever since the 1500s, when an unknown printer took a galley of type and scrambled it to make a type specimen book. It has survived not only five centuries, but also the leap into electronic typesetting, remaining essentially unchanged. It was popularised in the 1960s with the release of Letraset sheets containing Lorem Ipsum passages, and more recently with desktop publishing software like Aldus

### Second Block

Lorem Ipsum is simply dummy text of the printing and typesetting industry. Lorem Ipsum has been the industry's standard dummy text ever since the 1500s, when an unknown printer took a galley of type and scrambled it to make a type specimen book.

### Third Block

Lorem Ipsum is simply dummy text of the printing and typesetting industry. Lorem Ipsum has been the industry's standard dummy text ever since the 1500s, when an unknown printer took a galley of type and scrambled it to make a type specimen book.

## 7. jQuery Mobile – Widgets

A widget is a small gadget or control of your jQuery mobile application. Widgets can be very handy as they allow you to put your favorite applications on your home screen in order to quickly access them.

Following table demonstrates the types of Widgets in detail.

| Sr. No. | Types & Description   |
|---------|---|
| 1       | <b><u>Buttons</u></b><br>It specifies clickable button that includes content like text or images.   |
| 2       | <b><u>Checkbox</u></b><br>Checkboxes are used when more than one option is required to be selected.   |
| 3       | <b><u>Radiobox</u></b><br>Radio buttons are used when out of many options, just one option is required to be selected.  |
| 4       | <b><u>Datepicker</u></b><br>It is focused on the input to open an interactive calendar in a small overlay.  |
| 5       | <b><u>Collapsible</u></b><br>Collapsible allows you to expand or collapse the content whenever clicked on it. It is very helpful for mobile device, which presents a brief content. |
| 6       | <b><u>Controlgroup</u></b><br>Controlgroups provide a set of buttons to specify a single block that looks like a navigation component.  |
| 7       | <b><u>Filterable</u></b><br>By using the <i>data-filter="true"</i> attribute, you can filter the children of any element.   |
| 8       | <b><u>Flipswitch</u></b><br>Flip Switch allows you to turn off/on or true/false the switch by clicking on it for boolean style input.   |
| 9       | <b><u>Listview</u></b><br>The purpose of listview component is to render complex and customized content in lists.   |

|    |   |
|----|---|
| 10 | <b><u>Loader</u></b><br>The jQuery Mobile provides different ways of loading states to an element.  |
| 11 | <b><u>Navbar</u></b><br>The navbar widget is a set of buttons which links you to other web pages or sections.                                       |
| 12 | <b><u>Panels</u></b><br>Panels are used to display the DOM components in the box.   |
| 13 | <b><u>Popups</u></b><br>Popup is a user interface that appears within a small window to display text, images, and other content.                    |
| 14 | <b><u>Rangeslider</u></b><br>Rangeslider widget provides you with a pair of handles allowing you to select a numeric value range.                   |
| 15 | <b><u>Selectmenu</u></b><br>A select menu provides various options in the form of dropdown list, from where a user can select one or more options.  |
| 16 | <b><u>Slider</u></b><br>Slider allows you to choose a value by sliding the handle of the slider.  |
| 17 | <b><u>Table</u></b><br>jQuery Mobile uses the table to represent the data in terms of rows and columns, i.e. displays the data in a tabular format. |
| 18 | <b><u>Tabs</u></b><br>The tabs widget is jQuery ui tabs widget's extension, which accepts all the methods and options.                              |
| 19 | <b><u>Textinput</u></b><br>The <input> tag is used to declare an input element, a control that allows the user to input data.                       |
| 20 | <b><u>Toolbar</u></b><br>The jQuery mobile toolbar widget allows you to create headers and footers.   |

## Buttons

---

### Description

It specifies a clickable button that includes content like text or images by using the class ***ui-btn***.

You can create buttons in different ways as shown in the following table.

| Sr. No. | Button Type & Description   |
|---------|---|
| 1       | <b><u>Basic Button</u></b><br>Create a simple basic button in the page.               |
| 2       | <b><u>Rounded corners to button</u></b><br>You can add rounded corners to the button. |
| 3       | <b><u>Grouped Buttons</u></b><br>Creates the group of buttons in the page.            |
| 4       | <b><u>Shadow Buttons</u></b><br>You can add the shadow for the button.                |
| 5       | <b><u>Inline Button</u></b><br>You can display the button as wide as its content.     |
| 6       | <b><u>Mini Button</u></b><br>Display the smaller buttons in the page.                 |
| 7       | <b><u>Back Button</u></b><br>You can create the back button in the page.              |
| 8       | <b><u>Icons Position</u></b><br>You can set the position of icons in the button.      |
| 9       | <b><u>Disabled Button</u></b><br>You can disable the buttons.                         |

### Basic Buttons

---

#### Description

Create a simple basic button in the page using the ***ui-btn*** class.

## Example

Following example demonstrates the use of basic button in the jQuery Mobile Framework.

```
<!DOCTYPE html>
<head>
<meta name="viewport" content="width=device-width, initial-scale=1">
<link rel="stylesheet"
href="https://code.jquery.com/mobile/1.4.5/jquery.mobile-1.4.5.min.css">
<script src="https://code.jquery.com/jquery-1.11.3.min.js"></script>
<script src="https://code.jquery.com/mobile/1.4.5/jquery.mobile-
1.4.5.min.js"></script>
</head>
<body>
<div data-role="page" id="pageone">
<div data-role="header">
<h2>Header</h2>
</div>
<div data-role="main" class="ui-content">
<a href="#" class="ui-btn">Using Anchor</a>
<button class="ui-btn">Using Button</button>
</div>
<div data-role="footer">
<h2>Footer</h2>
</div>
</div>
</body>
</html>
```

## Output

Let's carry out the following steps to see how the above code works:

- Save the above html code as **btn\_basic\_button.html** file in your server root folder.
- Open this HTML file as [http://localhost/](http://localhost(btn_basic_button.html)btn\_basic\_button.html and the following output will be displayed.

## Header

[Using Anchor](#) [Using Button](#)

## Footer

## Rounded Buttons

---

### Description

You can add rounded corners to the buttons in the page using ***ui-corner-all*** class.

### Example

Following example demonstrates the use of rounded corner button in the jQuery Mobile Framework.

```
<!DOCTYPE html>
<head>
<meta name="viewport" content="width=device-width, initial-scale=1">
<link rel="stylesheet"
href="https://code.jquery.com/mobile/1.4.5/jquery.mobile-1.4.5.min.css">
<script src="https://code.jquery.com/jquery-1.11.3.min.js"></script>
<script src="https://code.jquery.com/mobile/1.4.5/jquery.mobile-
1.4.5.min.js"></script>
</head>
<body>
<div data-role="page" id="pageone">
<div data-role="header">
<h2>Header</h2>
</div>
<div data-role="main" class="ui-content">
<a href="#" class="ui-btn ui-corner-all">Rounded Anchor Element</a>
<button class="ui-btn ui-corner-all">Rounded Button Element</button>
</div>
<div data-role="footer">
<h2>Footer</h2>

```

```

    </div>
</div>
</body>
</html>
```

## Output

Let's carry out the following steps to see how the above code works:

- Save above html code as **btn\_rounded\_button.html** file in your server root folder.
- Open this HTML file as [http://localhost/](http://localhost(btn_rounded_button.html)btn\_rounded\_button.html and the following output will be displayed.

## Header

[Rounded Anchor Element](#) [Rounded Button Element](#)

## Footer

## Grouped Buttons

---

### Description

Create a group of buttons vertically and horizontally in the page using the **data-role="controlgroup"** attribute.

### Example

Following example demonstrates the use of grouping button in the jQuery Mobile Framework.

```

<!DOCTYPE html>
<head>
<meta name="viewport" content="width=device-width, initial-scale=1">
<link rel="stylesheet"
href="https://code.jquery.com/mobile/1.4.5/jquery.mobile-1.4.5.min.css">
<script src="https://code.jquery.com/jquery-1.11.3.min.js"></script>
```

```

<script src="https://code.jquery.com/mobile/1.4.5/jquery.mobile-
1.4.5.min.js"></script>
</head>
<body>
<div data-role="page" id="pageone">
    <div data-role="header">
        <h2>Header</h2>
    </div>
    <div data-role="main" class="ui-content">
        <div data-role="controlgroup" data-type="horizontal">
            <p>Horizontal Button Group</p>
            <a href="#" class="ui-btn">Apple</a>
            <a href="#" class="ui-btn">Orange</a>
            <a href="#" class="ui-btn">Grapes</a>
        </div><br>

        <div data-role="controlgroup" data-type="vertical">
            <p>Vertical Button Group</p>
            <a href="#" class="ui-btn">Apple</a>
            <a href="#" class="ui-btn">Orange</a>
            <a href="#" class="ui-btn">Grapes</a>
        </div>
    </div>
    <div data-role="footer">
        <h2>Footer</h2>
    </div>
</div>
</body>
</html>

```

## Output

Let's carry out the following steps to see how the above code works:

- Save the above html code as **btn\_grouped\_button.html** file in your server root folder.
- Open this HTML file as [http://localhost/](http://localhost(btn_grouped_button.html)btn\_grouped\_button.html and the following output will be displayed.

# Header

Horizontal Button Group

[Apple](#) [Orange](#) [Grapes](#)

Vertical Button Group

[Apple](#) [Orange](#) [Grapes](#)

# Footer

## Shadow Buttons

---

### Description

You can add the shadow for the button using the ***ui-shadow*** class.

### Example

Following example demonstrates the use of shadow button in the jQuery Mobile Framework.

```
<!DOCTYPE html>
<head>
<meta name="viewport" content="width=device-width, initial-scale=1">
<link rel="stylesheet"
href="https://code.jquery.com/mobile/1.4.5/jquery.mobile-1.4.5.min.css">
<script src="https://code.jquery.com/jquery-1.11.3.min.js"></script>
<script src="https://code.jquery.com/mobile/1.4.5/jquery.mobile-
1.4.5.min.js"></script>
</head>
<body>
<div data-role="page" id="pageone">
<div data-role="header">
<h2>Header</h2>
</div>
```

```

<div data-role="main" class="ui-content">
    <a href="#" class="ui-btn ui-shadow">Using Anchor Element</a>
    <button class="ui-btn ui-shadow">Using Button Element</button>
</div>
<div data-role="footer">
    <h2>Footer</h2>
</div>
</div>
</body>
</html>

```

## Output

Let's carry out the following steps to see how the above code works:

- Save the above html code as **btn\_shadow\_button.html** file in your server root folder.
- Open this HTML file as [http://localhost/](http://localhost(btn_shadow_button.html)btn\_shadow\_button.html and the following output will be displayed.

The screenshot shows a web page with the following structure:

- Header**: Contains two buttons: "Using Anchor Element" (blue link) and "Using Button Element" (gray button).
- Footer**: Contains the text "Footer".

## Inline Buttons

---

### Description

You can display the button as wide as its content using **ui-btn-inline** class.

### Example

Following example demonstrates the use of inline button in the jQuery Mobile Framework.

```
<!DOCTYPE html>
<head>
<meta name="viewport" content="width=device-width, initial-scale=1">
<link rel="stylesheet"
href="https://code.jquery.com/mobile/1.4.5/jquery.mobile-1.4.5.min.css">
<script src="https://code.jquery.com/jquery-1.11.3.min.js"></script>
<script src="https://code.jquery.com/mobile/1.4.5/jquery.mobile-
1.4.5.min.js"></script>
</head>
<body>
<div data-role="page" id="page1">
<div data-role="header">
<h2>First Page Header</h2>
</div>
<div data-role="main" class="ui-content">
<p>Normal Button</p>
<a href="#page2" class="ui-btn">Click here to go Page Two</a>
<p>Inline Button</p>
<a href="#page2" class="ui-btn ui-btn-inline">Click here to go Page Two</a>
</div>
<div data-role="footer">
<h2>First Page Footer</h2>
</div>
</div>

<div data-role="page" id="page2">
<div data-role="header">
<h2>Second Page Header</h2>
</div>
<div data-role="main" class="ui-content">
<p>These are inline buttons</p>
<a href="#page1" class="ui-btn ui-btn-inline">Click here to go Page One</a>
<a href="#page1" class="ui-btn ui-btn-inline">Click here to go to Page One</a>
<a href="#page1" class="ui-btn ui-btn-inline">Click here to go to Page One</a>
</div>
<div data-role="footer">
<h2>Second Page Footer</h2>
```

```
</div>
</div>
</body>
</html>
```

## Output

Let's carry out the following steps to see how the above code works:

- Save the above html code in btn\_inline\_button.html file.
- Open this HTML file in a browser using localhost, and the following output will be displayed.

## First Page Header

Normal Button

[Click here to go Page Two](#)

Inline Button

[Click here to go Page Two](#)

## First Page Footer

## Second Page Header

These are inline buttons

[Click here to go Page One](#) [Click here to go to Page One](#) [Click here to go to Page One](#)

## Second Page Footer

## Mini Button

---

### Description

Display the smaller button in the page using the ***ui-mini*** class.

### Example

Following example demonstrates the use of mini button in the jQuery Mobile Framework.

```
<!DOCTYPE html>
<head>
<meta name="viewport" content="width=device-width, initial-scale=1">
<link rel="stylesheet"
href="https://code.jquery.com/mobile/1.4.5/jquery.mobile-1.4.5.min.css">
<script src="https://code.jquery.com/jquery-1.11.3.min.js"></script>
<script src="https://code.jquery.com/mobile/1.4.5/jquery.mobile-1.4.5.min.js"></script>
</head>
<body>
<div data-role="page" id="page1">
<div data-role="header">
<h2>Header</h2>
</div>
<div data-role="main" class="ui-content">
<a href="#" class="ui-btn ui-mini">Small Button</a>
<a href="#" class="ui-btn">Normal Button</a>
</div>
<div data-role="footer">
<h2>Footer</h2>
</div>
</div>
</body>
</html>
```

### Output

Let's carry out the following steps to see how the above code works:

- Save the above html code as **btn\_mini\_button.html** file in your server root folder.
- Open this HTML file as [http://localhost/](http://localhost(btn_mini_button.html)btn\_mini\_button.html and the following output will be displayed.

## Header

[Small Button](#) [Normal Button](#)

## Footer

## Back Button

---

### Description

Create the back button in a page using the ***data-rel="back"*** attribute.

### Example

Following example demonstrates the use of back button in the jQuery Mobile Framework.

```
<!DOCTYPE html>
<head>
<meta name="viewport" content="width=device-width, initial-scale=1">
<link rel="stylesheet"
href="https://code.jquery.com/mobile/1.4.5/jquery.mobile-1.4.5.min.css">
<script src="https://code.jquery.com/jquery-1.11.3.min.js"></script>
<script src="https://code.jquery.com/mobile/1.4.5/jquery.mobile-
1.4.5.min.js"></script>
</head>
<body>
<div data-role="page" id="page1">
<div data-role="header">
<h2>Header</h2>
</div>

<div data-role="main" class="ui-content">
<a href="#page2" class="ui-btn">Go to Page Two</a>
</div>

```

```
<div data-role="footer">
  <h2>Footer</h2>
</div>
</div>

<div data-role="page" id="page2">
  <div data-role="header">
    <h2>Header</h2>
  </div>

  <div data-role="main" class="ui-content">
    <a href="#" class="ui-btn" data-rel="back">Go Back to Page One</a>
  </div>

  <div data-role="footer">
    <h2>Footer</h2>
  </div>
</div>
</body>
</html>
```

## Output

Let's carry out the following steps to see how the above code works:

- Save the above html code as **btn\_back\_button.html** file in your server root folder.
- Open this HTML file as [http://localhost/btn\\_back\\_button.html](http://localhost/btn_back_button.html) and the following output will be displayed.

## Header

[Go to Page Two](#)

## Footer

## Header

[Go Back to Page One](#)

## Footer

## Icon Position

---

### Description

You can display the position of icons in the button using ***ui-btn-icon-ico\_pos*** class.

### Example

Following example demonstrates the use of icon position in the button in the jQuery Mobile Framework.

```
<!DOCTYPE html>
<head>
<meta name="viewport" content="width=device-width, initial-scale=1">
<link rel="stylesheet"
href="https://code.jquery.com/mobile/1.4.5/jquery.mobile-1.4.5.min.css">
<script src="https://code.jquery.com/jquery-1.11.3.min.js"></script>
<script src="https://code.jquery.com/mobile/1.4.5/jquery.mobile-
1.4.5.min.js"></script>
</head>
<body>
<div data-role="page" id="page1">
<div data-role="header">
<h2>Header</h2>
</div>
<div data-role="main" class="ui-content">
<a href="#" class="ui-btn ui-icon-delete ui-btn-icon-left">Left Icon</a>
```

```

<a href="#" class="ui-btn ui-icon-delete ui-btn-icon-right">Right Icon</a>
<a href="#" class="ui-btn ui-icon-delete ui-btn-icon-top">Top Icon</a>
<a href="#" class="ui-btn ui-icon-delete ui-btn-icon-bottom">Bottom Icon</a>
</div>
<div data-role="footer">
    <h2>Footer</h2>
</div>
</div>
</body>
</html>

```

## Output

Let's carry out the following steps to see how the above code works:

- Save the above html code as **btn\_icon\_position.html** file in your server root folder.
- Open this HTML file as [http://localhost/](http://localhost(btn_icon_position.html) and the following output will be displayed.

**Header**

[Left Icon](#) [Right Icon](#) [Top Icon](#) [Bottom Icon](#)

**Footer**

## Disabled Button

---

### Description

You can disable the button using the **ui-state-disabled** class in the anchor element and using **disabled** attribute in the button element.

### Example

Following example demonstrates the use of disabled button in the jQuery Mobile Framework.

```
<!DOCTYPE html>
```

```
<head>
<meta name="viewport" content="width=device-width, initial-scale=1">
<link rel="stylesheet"
href="https://code.jquery.com/mobile/1.4.5/jquery.mobile-1.4.5.min.css">
<script src="https://code.jquery.com/jquery-1.11.3.min.js"></script>
<script src="https://code.jquery.com/mobile/1.4.5/jquery.mobile-
1.4.5.min.js"></script>
</head>
<body>
<div data-role="page" id="page1">
<div data-role="header">
<h2>Header</h2>
</div>
<div data-role="main" class="ui-content">
<a href="#" class="ui-btn ui-state-disabled">Disabled button using anchor
element</a>
<button disabled="">Disabled button using disabled attribute</button>
</div>
<div data-role="footer">
<h2>Footer</h2>
</div>
</div>
</body>
</html>
```

## Output

Let's carry out the following steps to see how the above code works:

- Save the above html code as **btn\_disabled\_button.html** file in your server root folder.
- Open this HTML file as [http://localhost/](http://localhost(btn_disabled_button.html)btn\_disabled\_button.html and the following output will be displayed.

## Header

[Disabled button using anchor element](#)

## Footer

### Checkbox Widget

#### Description

Checkboxes are used when more than one option is required to be selected. They are created using **<input>** tag but the type attribute is set to checkbox.

Following table demonstrates the types of checkbox in detail.

| Sr. No. | Types & Description  |
|---------|--|
| 1       | <b>Basic markup</b><br>Checkbox can be created using the <code>&lt;input&gt;</code> element with a <code>type="checkbox"</code> attribute and corresponding label.             |
| 2       | <b>Mini size</b><br>In an element, include attribute <code>data-mini="true"</code> for creating a mini version.  |
| 3       | <b>Vertical group</b><br>By default, the checkbox is set vertically.   |
| 4       | <b>Horizontal group</b><br>Set the checkbox horizontally by including the <code>data-type="horizontal"</code> to the <code>fieldset</code>                                     |
| 5       | <b>Icon position</b><br>The position of the checkbox icon button can be set at the right side using <code>data-iconpos="right"</code> attribute to the <code>fieldset</code> . |
| 6       | <b>Theme</b><br>Theme is set to the checkbox using attribute <code>data-theme</code> .   |

|   |   |
|---|---|
| 7 | <b>Disabled</b><br>Set the attribute <i>disabled=""</i> in the input to disable the checkbox.           |
| 8 | <b>Enhanced</b><br>The checkbox is enhanced using the <i>data-enhanced="true"</i> into the input field. |

## Checkbox Basic markup

---

### Description

Checkbox can be created by using the `<input>` element with a ***type="checkbox"*** attribute and a corresponding label. Include the input element inside the label element so that they are semantically associated.

### Example

Following example demonstrates the use of basic checkbox in the jQuery Mobile.

```
<!DOCTYPE html>
<html>
<head>
    <meta name="viewport" content="width=device-width, initial-scale=1">
    <link rel="stylesheet"
    href="https://code.jquery.com/mobile/1.4.5/jquery.mobile-1.4.5.min.css">
    <script src="https://code.jquery.com/jquery-1.11.3.min.js"></script>
    <script src="https://code.jquery.com/mobile/1.4.5/jquery.mobile-
    1.4.5.min.js"></script>
</head>
<body>
    <div data-role="page">
        <form>
            <label>
                <input type="checkbox">CheckBox 1
            </label>
            <label>
                <input type="checkbox">CheckBox 2
            </label>
            <label>
                <input type="checkbox">CheckBox 3
            </label>
        </form>
    </div>
</body>
</html>
```

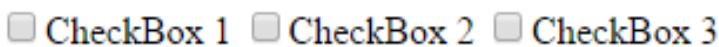
```

</div>
</body>
</html>
```

## Output

Let's carry out the following steps to see how the above code works:

- Save the above html code as **checkbox\_basic.html** file in your server root folder.
- Open this HTML file as [http://localhost/checkbox\\_basic.html](http://localhost/checkbox_basic.html) and the following output will be displayed.



## Checkbox Mini Size

---

### Description

In an element include attribute ***data-mini="true"*** for creating a mini version. It is useful in toolbars and tight spaces for more compact version.

### Example

Following example demonstrates the use of mini size checkbox in the jQuery Mobile.

```

<!DOCTYPE html>
<html>
<head>
  <meta name="viewport" content="width=device-width, initial-scale=1">
  <link rel="stylesheet"
  href="https://code.jquery.com/mobile/1.4.5/jquery.mobile-1.4.5.min.css">
  <script src="https://code.jquery.com/jquery-1.11.3.min.js"></script>
  <script src="https://code.jquery.com/mobile/1.4.5/jquery.mobile-
  1.4.5.min.js"></script>
</head>
<body>
  <div data-role="page">
    <form>
```

```

<input type="checkbox" id="mini-1" data-mini="true">
<label for="mini-1">Checkbox mini 1</label>

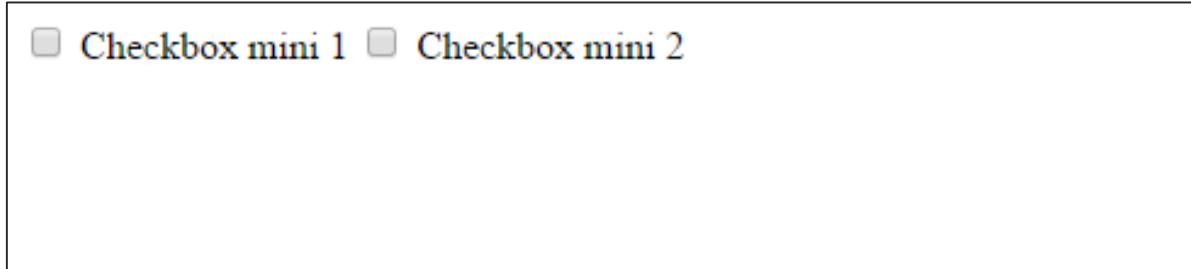
<input type="checkbox" id="mini-2" data-mini="true">
<label for="mini-2">Checkbox mini 2</label>
</form>
</div>
</body>
</html>

```

## Output

Let's carry out the following steps to see how the above code works:

- Save the above html code as **mini\_size.html** file in your server root folder.
- Open this HTML file as [http://localhost/mini\\_size.html](http://localhost/mini_size.html) and the following output will be displayed.



Checkbox mini 1   Checkbox mini 2

## Checkbox Vertical Group

---

### Description

Multiple checkbox is grouped by setting the ***data-role="controlgroup"*** attribute in the *fieldset*. Automatically, the framework will set the corners round at the bottom and top corner, and it will remove all margins. By default, the checkbox is set vertically.

### Example

Following example demonstrates the use of vertical group checkbox in the jQuery Mobile.

```

<!DOCTYPE html>
<html>
<head>
<meta name="viewport" content="width=device-width, initial-scale=1">

```

```

<link rel="stylesheet"
href="https://code.jquery.com/mobile/1.4.5/jquery.mobile-1.4.5.min.css">
<script src="https://code.jquery.com/jquery-1.11.3.min.js"></script>
<script src="https://code.jquery.com/mobile/1.4.5/jquery.mobile-
1.4.5.min.js"></script>
</head>
<body>
<h2>Vertical Group Example</h2>
<form>
  <fieldset data-role="controlgroup">
    <input type="checkbox" id="checkbox 1">
    <label for="checkbox 1">Checkbox 1</label>

    <input type="checkbox" id="checkbox 2">
    <label for="checkbox 2">Checkbox 2</label>

    <input type="checkbox" id="checkbox 3">
    <label for="checkbox 3">Checkbox 3</label>
  </fieldset>
</form>
</body>
</html>

```

## Output

Let's carry out the following steps to see how the above code works:

- Save the above html code as **vertical\_group.html** file in your server root folder.
- Open this HTML file as [http://localhost/vertical\\_group.html](http://localhost/vertical_group.html) and the following output will be displayed.

## Vertical Group Example

Checkbox 1  Checkbox 2  Checkbox 3

## Checkbox Horizontal Group

---

### Description

Set the checkbox horizontally by including the ***data-type="horizontal"*** to the *fieldset*. You can select the checkbox button more than one at a time.

### Example

Following example demonstrates the use of horizontal group checkbox in the jQuery Mobile.

```
<!DOCTYPE html>
<html>
<head>
    <meta name="viewport" content="width=device-width, initial-scale=1">
    <link rel="stylesheet"
    href="https://code.jquery.com/mobile/1.4.5/jquery.mobile-1.4.5.min.css">
    <script src="https://code.jquery.com/jquery-1.11.3.min.js"></script>
    <script src="https://code.jquery.com/mobile/1.4.5/jquery.mobile-
1.4.5.min.js"></script>
</head>
<body>
    <h2>Horizontal Group Example</h2>
    <form>
        <fieldset data-role="controlgroup" data-type="horizontal">
            <input type="checkbox" id="checkbox 1">
            <label for="checkbox 1">Checkbox 1</label>

            <input type="checkbox" id="checkbox 2">
            <label for="checkbox 2">Checkbox 2</label>

            <input type="checkbox" id="checkbox 3">
            <label for="checkbox 3">Checkbox 3</label>
        </fieldset>
    </form>
</body>
</html>
```

### Output

Let's carry out the following steps to see how the above code works:

- Save the above html code as **horizontal\_group.html** file in your server root folder.
- Open this HTML file as [http://localhost/horizontal\\_group.html](http://localhost/horizontal_group.html) and the following output will be displayed.

## Horizontal Group Example

Checkbox 1  Checkbox 2  Checkbox 3

## Checkbox Icon Position

---

### Description

The position of the checkbox icon button can be set at the right side using **data-iconpos="right"** attribute to the *fieldset*. By default, the checkbox icon is set at on the left side.

### Example

Following example demonstrates the use of icon position checkbox in the jQuery Mobile.

```
<!DOCTYPE html>
<html>
<head>
    <meta name="viewport" content="width=device-width, initial-scale=1">
    <link rel="stylesheet"
    href="https://code.jquery.com/mobile/1.4.5/jquery.mobile-1.4.5.min.css">
    <script src="https://code.jquery.com/jquery-1.11.3.min.js"></script>
    <script src="https://code.jquery.com/mobile/1.4.5/jquery.mobile-
    1.4.5.min.js"></script>
</head>
<body>
    <h2>Icon Position Example</h2>
    <form>
        <fieldset data-role="controlgroup" data-iconpos="right">
```

```

<input type="checkbox" id="checkbox 1">
<label for="checkbox 1">Checkbox 1</label>

<input type="checkbox" id="checkbox 2">
<label for="checkbox 2">Checkbox 2</label>

<input type="checkbox" id="checkbox 3">
<label for="checkbox 3">Checkbox 3</label>
</fieldset>
</form>
</body>
</html>

```

## Output

Let's carry out the following steps to see how the above code works:

- Save the above html code as **icon\_position.html** file in your server root folder.
- Open this HTML file as [http://localhost/icon\\_position.html](http://localhost/icon_position.html) and the following output will be displayed.

## Icon Position Example



Checkbox 1   Checkbox 2   Checkbox 3

## Checkbox Theme

---

### Description

Theme is set to the checkbox using attribute ***data-theme*** on the fieldset or to a particular checkbox input.

### Example

Following example demonstrates the use of checkbox theme in the jQuery Mobile.

```
<!DOCTYPE html>
```

```

<html>
<head>
    <meta name="viewport" content="width=device-width, initial-scale=1">
    <link rel="stylesheet"
    href="https://code.jquery.com/mobile/1.4.5/jquery.mobile-1.4.5.min.css">
    <script src="https://code.jquery.com/jquery-1.11.3.min.js"></script>
    <script src="https://code.jquery.com/mobile/1.4.5/jquery.mobile-
1.4.5.min.js"></script>
</head>
<body>
    <h2>Checkbox Theme Example</h2>
    <form>
        <fieldset data-role="controlgroup">
            <input type="checkbox" id="checkbox 1" data-theme="b">
            <label for="checkbox 1">Checkbox 1</label>

            <input type="checkbox" id="checkbox 2" data-theme="b">
            <label for="checkbox 2">Checkbox 2</label>

            <input type="checkbox" id="checkbox 3" data-theme="b">
            <label for="checkbox 3">Checkbox 3</label>
        </fieldset>
    </form>
</body>
</html>

```

## Output

Let's carry out the following steps to see how the above code works:

- Save the above html code as **theme.html** file in your server root folder.
- Open this HTML file as <http://localhost/theme.html> and the following output will be displayed.

## Checkbox Theme Example

Checkbox 1  Checkbox 2  Checkbox 3

## Checkbox Disabled

### Description

Set the attribute **disabled=""** in the input to disable the checkbox.

### Example

Following example demonstrates the use of disabled checkbox theme in the jQuery Mobile.

```
<!DOCTYPE html>
<html>
<head>
    <meta name="viewport" content="width=device-width, initial-scale=1">
    <link rel="stylesheet"
    href="https://code.jquery.com/mobile/1.4.5/jquery.mobile-1.4.5.min.css">
    <script src="https://code.jquery.com/jquery-1.11.3.min.js"></script>
    <script src="https://code.jquery.com/mobile/1.4.5/jquery.mobile-
    1.4.5.min.js"></script>
</head>
<body>
    <h2>Disabled Checkbox Example</h2>
    <form>
        <fieldset data-role="controlgroup">
            <input disabled="" type="checkbox" id="checkbox 1" data-theme="b">
            <label for="checkbox 1">Checkbox 1</label>

            <input disabled="" type="checkbox" id="checkbox 2" data-theme="b">
            <label for="checkbox 2">Checkbox 2</label>
    
```

```

</fieldset>
</form>
</body>
</html>

```

## Output

Let's carry out the following steps to see how the above code works:

- Save the above html code as **disabled.html** file in your server root folder.
- Open this HTML file as <http://localhost/disabled.html> and the following output will be displayed.

## Disabled Checkbox Example



Checkbox 1  Checkbox 2

## Checkbox Enhanced

---

### Description

The checkbox is enhanced using the ***data-enhanced="true"*** into the input field.

### Example

Following example demonstrates the use of enhanced checkbox in the jQuery Mobile.

```

<!DOCTYPE html>
<html>
<head>
    <meta name="viewport" content="width=device-width, initial-scale=1">
    <link rel="stylesheet"
    href="https://code.jquery.com/mobile/1.4.5/jquery.mobile-1.4.5.min.css">
    <script src="https://code.jquery.com/jquery-1.11.3.min.js"></script>
    <script src="https://code.jquery.com/mobile/1.4.5/jquery.mobile-
    1.4.5.min.js"></script>

```

```

</head>
<body>
  <h2>Enhanced Checkbox Example</h2>
  <form>
    <div class="ui-checkbox">
      <label for="checkbox1" class="ui-btn ui-corner-all">Checkbox</label>
      <input type="checkbox" id="checkbox1" data-enhanced="true">
    </div>
  </form>
</body>
</html>

```

## Output

Let's carry out the following steps to see how the above code works:

- Save the above html code as **enhanced.html** file in your server root folder.
- Open this HTML file as <http://localhost/enhanced.html> and the following output will be displayed.

## Enhanced Checkbox Example

Checkbox

## Radio Widget

---

### Description

Radio buttons are used when out of many options, just one option is required to be selected. They are also created using **<input>** tag but *type* attribute is set to *radio*.

Following table demonstrates the types of radio button in detail.

| Sr. No. | Types & Description   |
|---------|---|
| 1       | <b><u>Basic markup</u></b><br>Radio buttons can be created using the <code>&lt;input&gt;</code> element with a <code>type="radio"</code> attribute and a corresponding label. |
| 2       | <b><u>Mini size</u></b><br>In fieldset element, include attribute <code>data-mini="true"</code> for creating a mini version.  |
| 3       | <b><u>Vertical group</u></b><br>By default, the radio is set vertically.  |
| 4       | <b><u>Horizontal group</u></b><br>Set the radio horizontally by including the <code>data-type="horizontal"</code> to the fieldset.  |
| 5       | <b><u>Icon position</u></b><br>The position of the radio icon button can be set at the right side using <code>data-iconpos="right"</code> attribute to the fieldset.          |
| 6       | <b><u>Theme</u></b><br>Theme is set to the radio button using the attribute <code>data-theme</code> .   |
| 7       | <b><u>Disabled</u></b><br>Set the attribute <code>disabled="disabled"</code> in the input to disable the radio button.  |
| 8       | <b><u>Enhanced</u></b><br>The radio is enhanced using the <code>data-enhanced="true"</code> into the input field.   |

## Basic Radio Widget

---

### Description

Radio buttons can be created using the `<input>` element with a `type="radio"` attribute and a corresponding label. Set the input element with **for** attribute to match the label id so that they are semantically associated.

### Example

Following example demonstrates the use of basic radio button in the jQuery Mobile.

```
<!DOCTYPE html>
<html>
```

```

<head>
    <meta name="viewport" content="width=device-width, initial-scale=1">
    <link rel="stylesheet"
    href="https://code.jquery.com/mobile/1.4.5/jquery.mobile-1.4.5.min.css">
    <script src="https://code.jquery.com/jquery-1.11.3.min.js"></script>
    <script src="https://code.jquery.com/mobile/1.4.5/jquery.mobile-
1.4.5.min.js"></script>
</head>
<body>
    <form>
        <h2>Basic Radio Example</h2>
        <fieldset data-role="controlgroup">
            <input type="radio" id="radio1" name="radio-choice-0">
            <label for="radio1">Radio 1</label>

            <input type="radio" id="radio2" name="radio-choice-0">
            <label for="radio2">Radio 2</label>

            <input type="radio" id="radio3" name="radio-choice-0">
            <label for="radio3">Radio 3</label>
        </fieldset>
    </form>
</body>
</html>

```

## Output

Let's carry out the following steps to see how the above code works:

- Save the above html code as **radio\_basic.html** file in your server root folder.
- Open this HTML file as [http://localhost/radio\\_basic.html](http://localhost/radio_basic.html) and the following output will be displayed.

## Basic Radio Example

Radio 1  Radio 2  Radio 3

## Basic Radio Mini Size Widget

### Description

In fieldset element, include attribute ***data-mini="true"*** for creating a mini version. It is useful in toolbars and tight spaces for more compact version.

### Example

Following example demonstrates the use of mini size radio button in the jQuery Mobile.

```
<!DOCTYPE html>
<html>
<head>
    <meta name="viewport" content="width=device-width, initial-scale=1">
    <link rel="stylesheet"
    href="https://code.jquery.com/mobile/1.4.5/jquery.mobile-1.4.5.min.css">
    <script src="https://code.jquery.com/jquery-1.11.3.min.js"></script>
    <script src="https://code.jquery.com/mobile/1.4.5/jquery.mobile-
1.4.5.min.js"></script>
</head>
<body>
<form>
    <h2>Radio Mini Size Example</h2>
    <fieldset data-role="controlgroup" data-mini="true">
        <input type="radio" id="radio1" name="radio-choice-0">
        <label for="radio1">Radio 1</label>

        <input type="radio" id="radio2" name="radio-choice-0">
        <label for="radio2">Radio 2</label>

        <input type="radio" id="radio3" name="radio-choice-0">
        <label for="radio3">Radio 3</label>
    </fieldset>
</form>
</body>
</html>
```

## Output

Let's carry out the following steps to see how the above code works:

- Save the above html code as **radio\_mini.html** file in your server root folder.
- Open this HTML file as [http://localhost/radio\\_mini.html](http://localhost/radio_mini.html) and the following output will be displayed.

### Radio Mini Size Example

Radio 1  Radio 2  Radio 3

## Radio Vertical Group Widget

---

### Description

Multiple radio buttons are integrated in the group by setting the **data-role="controlgroup"** attribute in the fieldset. Automatically, the framework will set the corners round at the bottom and top corner, and remove all margins. By default, the radio is set vertically.

### Example

Following example demonstrates the use of radio button vertically in the jQuery Mobile.

```
<!DOCTYPE html>
<html>
<head>
    <meta name="viewport" content="width=device-width, initial-scale=1">
    <link rel="stylesheet"
    href="https://code.jquery.com/mobile/1.4.5/jquery.mobile-1.4.5.min.css">
    <script src="https://code.jquery.com/jquery-1.11.3.min.js"></script>
    <script src="https://code.jquery.com/mobile/1.4.5/jquery.mobile-
    1.4.5.min.js"></script>
</head>
<body>
    <form>
        <h2>Radio Vertical Group Example</h2>
```

```

<fieldset data-role="controlgroup">
    <input type="radio" id="radio1" name="radio-choice-0" checked="checked">
    <label for="radio1">Radio 1</label>

    <input type="radio" id="radio2" name="radio-choice-0">
    <label for="radio2">Radio 2</label>

    <input type="radio" id="radio3" name="radio-choice-0">
    <label for="radio3">Radio 3</label>
</fieldset>
</form>
</body>
</html>

```

## Output

Let's carry out the following steps to see how the above code works:

- Save the above html code as **radio\_vertical\_group.html** file in your server root folder.
- Open this HTML file as [http://localhost/radio\\_vertical\\_group.html](http://localhost/radio_vertical_group.html) and the following output will be displayed.

## Radio Vertical Group Example

Radio 1 Radio 2 Radio 3

## Radio Horizontal Group Widget

---

### Description

Set the radio horizontally by including the ***data-type="horizontal"*** to the fieldset.

## Example

Following example demonstrates the use of radio button horizontally in the jQuery Mobile.

```
<!DOCTYPE html>
<html>
<head>
    <meta name="viewport" content="width=device-width, initial-scale=1">
    <link rel="stylesheet"
    href="https://code.jquery.com/mobile/1.4.5/jquery.mobile-1.4.5.min.css">
    <script src="https://code.jquery.com/jquery-1.11.3.min.js"></script>
    <script src="https://code.jquery.com/mobile/1.4.5/jquery.mobile-
    1.4.5.min.js"></script>
</head>
<body>
    <form>
        <h2>Radio Horizontal Group Example</h2>
        <fieldset data-role="controlgroup" data-type="horizontal">
            <input type="radio" id="radio1" name="radio-choice-0" checked="checked">
            <label for="radio1">Radio 1</label>

            <input type="radio" id="radio2" name="radio-choice-0">
            <label for="radio2">Radio 2</label>

            <input type="radio" id="radio3" name="radio-choice-0">
            <label for="radio3">Radio 3</label>
        </fieldset>
    </form>
</body>
</html>
```

## Output

Let's carry out the following steps to see how the above code works:

- Save the above html code as ***radio\_horizontal.html*** file in your server root folder.

- Open this HTML file as [http://localhost/radio\\_horizontal.html](http://localhost/radio_horizontal.html) and the following output will be displayed.

## Radio Horizontal Group Example

Radio 1  Radio 2  Radio 3

## Radio Icon Position Widget

### Description

The position of the radio icon button can be set at the right side using **data-iconpos="right"** attribute to the fieldset. By default, the radio icon is set on the left side.

### Example

Following example demonstrates the use of radio button icon position in the jQuery Mobile.

```
<!DOCTYPE html>
<html>
<head>
    <meta name="viewport" content="width=device-width, initial-scale=1">
    <link rel="stylesheet"
    href="https://code.jquery.com/mobile/1.4.5/jquery.mobile-1.4.5.min.css">
    <script src="https://code.jquery.com/jquery-1.11.3.min.js"></script>
    <script src="https://code.jquery.com/mobile/1.4.5/jquery.mobile-
    1.4.5.min.js"></script>
</head>
<body>
<form>
    <h2>Radio Icon Position Example</h2>
    <fieldset data-role="controlgroup" data-iconpos="right">
        <input type="radio" id="radio1" name="radio-choice-0" checked="checked">
        <label for="radio1">Radio 1</label>

        <input type="radio" id="radio2" name="radio-choice-0">
    
```

```

<label for="radio2">Radio 2</label>

<input type="radio" id="radio3" name="radio-choice-0">
<label for="radio3">Radio 3</label>
</fieldset>
</form>
</body>
</html>

```

## Output

Let's carry out the following steps to see how the above code works:

- Save the above html code as **radio\_icon.html** file in your server root folder.
- Open this HTML file as [http://localhost/radio\\_icon.html](http://localhost/radio_icon.html) and the following output will be displayed.

## Radio Icon Position Example

## Radio Widget Theme

---

### Description

Theme is set to the radio button using attribute ***data-theme*** on the fieldset or to a particular radio input.

### Example

Following example demonstrates the use of radio theme in the jQuery Mobile.

```

<!DOCTYPE html>
<html>
<head>
    <meta name="viewport" content="width=device-width, initial-scale=1">

```

```
<link rel="stylesheet"
href="https://code.jquery.com/mobile/1.4.5/jquery.mobile-1.4.5.min.css">
<script src="https://code.jquery.com/jquery-1.11.3.min.js"></script>
<script src="https://code.jquery.com/mobile/1.4.5/jquery.mobile-
1.4.5.min.js"></script>
</head>
<body>
<form>
    <h2>Radio Theme Example</h2>
    <fieldset data-role="controlgroup" data-theme="b">
        <input type="radio" id="radio1" name="radio-choice-0"
checked="checked">
        <label for="radio1">Radio 1</label>

        <input type="radio" id="radio2" name="radio-choice-0">
        <label for="radio2">Radio 2</label>

        <input type="radio" id="radio3" name="radio-choice-0">
        <label for="radio3">Radio 3</label>
    </fieldset>
</form>
</body>
</html>
```

## Output

Let's carry out the following steps to see how the above code works:

- Save the above html code as **radio\_theme.html** file in your server root folder.
- Open this HTML file as [http://localhost/radio\\_theme.html](http://localhost/radio_theme.html) and the following output will be displayed.

## Radio Theme Example

Radio 1  Radio 2  Radio 3

## Radio Disabled Widget

### Description

Set the attribute **disabled="disabled"** in the input to disable the radio button.

### Example

Following example demonstrates the use of disable the radio button in the jQuery Mobile.

```
<!DOCTYPE html>
<html>
<head>
    <meta name="viewport" content="width=device-width, initial-scale=1">
    <link rel="stylesheet"
    href="https://code.jquery.com/mobile/1.4.5/jquery.mobile-1.4.5.min.css">
    <script src="https://code.jquery.com/jquery-1.11.3.min.js"></script>
    <script src="https://code.jquery.com/mobile/1.4.5/jquery.mobile-
    1.4.5.min.js"></script>
</head>
<body>
    <form>
        <h2>Radio Disable Example</h2>
        <fieldset data-role="controlgroup" data-theme="b">
            <input type="radio" id="radio1" name="radio-choice-0" disabled="disabled">
            <label for="radio1">Radio 1</label>

            <input type="radio" id="radio2" name="radio-choice-0" disabled="disabled">
            <label for="radio2">Radio 2</label>

            <input type="radio" id="radio3" name="radio-choice-0" disabled="disabled">
            <label for="radio3">Radio 3</label>
        </fieldset>
    </form>
</body>
</html>
```

```

</fieldset>
</form>
</body>
</html>

```

## Output

Let's carry out the following steps to see how the above code works:

- Save the above html code as **radio\_disable.html** file in your server root folder.
- Open this HTML file as [http://localhost/radio\\_disable.html](http://localhost/radio_disable.html) and the following output will be displayed.

## Radio Disable Example

Radio 1 Radio 2 Radio 3

## Radio Enhanced Widget

### Description

The radio is enhanced using the attribute ***data-enhanced="true"*** in the input field.

### Example

Following example demonstrates the use of radio enhanced in the jQuery Mobile.

```

<!DOCTYPE html>
<html>
<head>
    <meta name="viewport" content="width=device-width, initial-scale=1">
    <link rel="stylesheet"
    href="https://code.jquery.com/mobile/1.4.5/jquery.mobile-1.4.5.min.css">
    <script src="https://code.jquery.com/jquery-1.11.3.min.js"></script>
    <script src="https://code.jquery.com/mobile/1.4.5/jquery.mobile-
    1.4.5.min.js"></script>
</head>

```

```

<body>
    <form>
        <h2>Radio Enhanced Example</h2>
        <div class="ui-radio">
            <label for="radio1" class="ui-btn">Radio 1</label>
            <input type="radio" id="radio1" name="radio-enhanced" data-enhanced="true">
        </div>

        <div class="ui-radio">
            <label for="radio2" class="ui-btn">Radio 2</label>
            <input type="radio" id="radio2" name="radio-enhanced" data-enhanced="true">
        </div>
    </form>
</body>
</html>

```

## Output

Let's carry out the following steps to see how the above code works:

- Save the above html code as **radio\_enhanced.html** file in your server root folder.
- Open this HTML file as [http://localhost/radio\\_enhanced.html](http://localhost/radio_enhanced.html) and the following output will be displayed.

## Radio Enhanced Example

Radio 1   
 Radio 2

## Datapicker Widget

---

### Description

The DatePicker function can be used with widgets using JqueryUI in jQuery mobile as it does not support jQuery mobile widget. It is used to focus on the input to open an interactive calendar in a small overlay.

## Popup Datapicker

The calendar pops up when input is focused to insert date. Add ***data-role="date"*** attribute in the <input> field to include the date, it displays in *dd/mm/yy* format.

### Example

Following example demonstrates the use of popup datapicker in the jQuery Mobile.

```
<!DOCTYPE html>
<html>
<head>
    <meta name="viewport" content="width=device-width, initial-scale=1">
    <link rel="stylesheet"
    href="https://code.jquery.com/mobile/1.4.5/jquery.mobile-1.4.5.min.css">
    <link rel="stylesheet" href="https://rawgithub.com/arschmitz/jquery-mobile-
    datepicker-wrapper/master/jquery.mobile.datepicker.css" />
    <script src="https://code.jquery.com/jquery-1.11.3.min.js"></script>
    <script src="https://code.jquery.com/mobile/1.4.5/jquery.mobile-
    1.4.5.min.js"></script>
    <script src="https://rawgithub.com/jquery/jquery-ui/1-10-
    stable/ui/jquery.ui.datepicker.js"></script>
    <script src="https://rawgithub.com/arschmitz/jquery-mobile-datepicker-
    wrapper/master/jquery.mobile.datepicker.js">
    </script>
</head>
<body>
<h2>Popup Datapicker Example</h2>
<form>
    <input type="text" data-role="date">
</form>
</body>
</html>
```

### Output

Let's carry out the following steps to see how the above code works:

- Save the above html code as **popup\_datapicker.html** file in your server root folder.

- Open this HTML file as `http://localhost/popup_datapicker.html` and the following output will be displayed.

## Popup Datapicker Example

### Inline Datapicker

Include `data-inline="true"` to display the interactive calendar.

### Example

Following example demonstrates the use of inline datapicker in the jQuery Mobile.

```
<!DOCTYPE html>
<html>
<head>
    <meta name="viewport" content="width=device-width, initial-scale=1">
    <link rel="stylesheet"
    href="https://code.jquery.com/mobile/1.4.5/jquery.mobile-1.4.5.min.css">
    <link rel="stylesheet" href="https://rawgithub.com/arschmitz/jquery-mobile-
    datepicker-wrapper/master/jquery.mobile.datepicker.css" />
    <script src="https://code.jquery.com/jquery-1.11.3.min.js"></script>
    <script src="https://code.jquery.com/mobile/1.4.5/jquery.mobile-
    1.4.5.min.js"></script>
    <script src="https://rawgithub.com/jquery/jquery-ui/1-10-
    stable/ui/jquery.ui.datepicker.js"></script>
    <script src="https://rawgithub.com/arschmitz/jquery-mobile-datepicker-
    wrapper/master/jquery.mobile.datepicker.js">
    </script>
</head>
<body>
<h2>Popup Datapicker Example</h2>
<form>
    <input type="text" data-role="date" data-inline="true">

```

```
</form>
</body>
</html>
```

## Output

Let's carry out the following steps to see how the above code works:

- Save the above html code as **inline\_datapicker.html** file in your server root folder.
- Open this HTML file as [http://localhost/inline\\_datapicker.html](http://localhost/inline_datapicker.html) and the following output will be displayed.

## Inline Datapicker Example



## Collapsible Widget

Collapsible allows you to expand or collapse the content whenever it is clicked. It is very helpful for mobile device, which presents a brief content.

Following table lists different types of collapsible widget used in jQuery mobile.

| Sr. No. | Collapsible Type & Description   |
|---------|--|
| 1       | <b><u>Collapsible</u></b><br>Collapsibles are the basic and easy widget to display short content.                                    |
| 2       | <b><u>Dynamic collapsible</u></b><br>Collapsible can be added to collapsible set dynamically.  |
| 3       | <b><u>Collapsible set</u></b><br>In jQuery Mobile, an accordion can be created by putting a series of separate collapsible into set. |
| 4       | <b><u>Nested Collapsible</u></b>   |

|  |   |
|--|---|
|  | Collapsible can be nested inside other collapsible content. |
|--|---|

## Collapsible

---

### Description

Collapsibles are the basic and easy widget to display short content. Following table lists the collapsible types used in jQuery Mobile.

| Sr. No. | Collapsible types & Description  |
|---------|--|
| 1       | <b><u>Basic collapsible</u></b><br>Collapsible block of content can be created using <i>data-role="collapsible"</i> attribute to a container.                                |
| 2       | <b><u>Theme</u></b><br>To set a collapsible theme, you can use <i>data-theme</i> attribute and add <i>data-content-theme</i> attribute with a theme swatch letter specified. |
| 3       | <b><u>Expanded</u></b><br>It enables you to expand the collapsible content on page load using <i>data-collapsed</i> attribute and setting it to <i>false</i> .               |
| 4       | <b><u>Mini sized</u></b><br>It can be used whenever you need a compact version of collapsible. This can be achieved by using <i>data-mini</i> and setting to <i>true</i> .   |
| 5       | <b><u>Icons</u></b><br>The collapsible heading's default icons can be overridden by using <i>data-collapsed-icon</i> and <i>data-expanded-icon</i> attributes.               |
| 6       | <b><u>Icon positioning</u></b><br>The collapsible heading's default icon positioning can be overridden by using <i>data-iconpos</i> attribute.                               |

|    |  |
|----|--|
| 7  | <b><u>Legend</u></b><br>You can display form elements in collapsible by using a <i>fieldset</i> and <i>legend</i> .  |
| 8  | <b><u>Non-inset collapsible</u></b><br>To remove inset appearance and to display the collapsible to its full width without corner styling, you can add the <i>data-inset="false"</i> attribute.          |
| 9  | <b><u>Set of individual collapsibles</u></b><br>You can apply a series of individual collapsibles and it allows you to open multiple rows at once.   |
| 10 | <b><u>Pre-rendered markup</u></b><br>To improve the load time of your page, you can provide pre-rendered markup for collapsible by adding <i>data-enhanced</i> attribute and setting it to <i>true</i> . |

## Basic Collapsible

---

### Description

Collapsible block of content can be created using ***data-role="collapsible"*** attribute to a container. Inside the container, you can add any header or legend element. Header will be styled by the framework to appear like a clickable button.

Below the header, you can add HTML markup of your choice and it will be shown/hidden whenever the heading is clicked.

### Example

Following example demonstrates the use of basic collapsible in jQuery Mobile.

```
<!DOCTYPE html>
<head>
    <title>Basic collapsible</title>
    <meta name="viewport" content="width=device-width, initial-scale=1">
    <link rel="stylesheet"
    href="https://code.jquery.com/mobile/1.4.5/jquery.mobile-1.4.5.min.css">
    <script src="https://code.jquery.com/jquery-1.11.3.min.js"></script>
    <script src="https://code.jquery.com/mobile/1.4.5/jquery.mobile-
1.4.5.min.js"></script>
</head>
<body>
    <div data-role="collapsible">
        <legend>Heading</legend>
        <p>Hi! This is collapsible content.</p>
    </div>
</body>
```

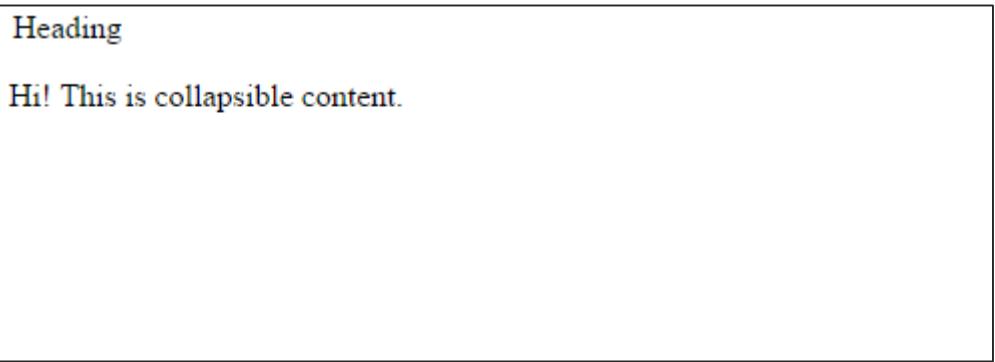
```

</div>
</body>
</html>
```

## Output

Let's carry out the following steps to see how the above code works:

- Save the above html code as **jquery\_mobile\_basic\_collapsible.html** file in your server root folder.
- Open this HTML file as [http://localhost/jquery\\_mobile\\_basic\\_collapsible.html](http://localhost/jquery_mobile_basic_collapsible.html) and the following output will be displayed.



The screenshot shows a mobile device screen. At the top, there is a blue header bar with the word "Heading" in white. Below the header, the main content area contains the text "Hi! This is collapsible content." in black. There is a small downward-pointing arrow icon located to the left of the text, indicating it is a collapsible element.

## Collapsible Theme

---

### Description

To set a collapsible theme, you can use ***data-theme*** attribute and add ***data-content-theme*** attribute with a theme swatch letter specified. You can set ***data-content-theme*** to *false* in order to avoid default inherit of collapsible content.

Default jQuery Mobile theme includes multiple swatches like ***a***, ***b***, ***c***, ***d*** and ***e***. A ***swatch*** is a unified color concept, which governs many things like color, background, text, and more. These swatches can be applied to individual objects.

### Example

Following example demonstrates the use of collapsible theme in jQuery Mobile.

```

<!DOCTYPE html>
<head>
    <title>Collapsible Theme</title>
    <meta name="viewport" content="width=device-width, initial-scale=1">
    <link rel="stylesheet"
        href="https://code.jquery.com/mobile/1.4.5/jquery.mobile-1.4.5.min.css">
```

```

<script src="https://code.jquery.com/jquery-1.11.3.min.js"></script>
<script src="https://code.jquery.com/mobile/1.4.5/jquery.mobile-
1.4.5.min.js"></script>
</head>
<body>
<div data-role="collapsible" data-theme="b" data-content-theme="b">
    <h4>Heading</h4>
    <p>Hi! This is collapsible content with a themed block of content set to
    "b".</p>
</div>

<div data-role="collapsible" data-content-theme="false">
    <h4>Heading</h4>
    <p>Hi! This is collapsible content without a theme.</p>
</div>
</body>
</html>

```

## Output

Let's carry out the following steps to see how the above code works:

- Save the above html code as **collapsible\_theme.html** file in your server root folder.
- Open this HTML file as [http://localhost/collapsible\\_theme.html](http://localhost/collapsible_theme.html) and the following output will be displayed.

|  |
|--|
| <b>Heading</b>   |
| Hi! This is collapsible content with a themed block of content set to "b". |
| <b>Heading</b>   |
| Hi! This is collapsible content without a theme.                           |

## Collapsible Expanded

---

### Description

You can expand the collapsible content on page load by setting ***data-collapsed*** attribute to *false*.

## Example

Following example demonstrates the use of expanded collapsible in jQuery Mobile.

```
<!DOCTYPE html>
<head>
    <title>Collapsible Expanded</title>
    <meta name="viewport" content="width=device-width, initial-scale=1">
    <link rel="stylesheet"
    href="https://code.jquery.com/mobile/1.4.5/jquery.mobile-1.4.5.min.css">
    <script src="https://code.jquery.com/jquery-1.11.3.min.js"></script>
    <script src="https://code.jquery.com/mobile/1.4.5/jquery.mobile-
1.4.5.min.js"></script>
</head>
<body>
    <div data-role="collapsible" data-collapsed="false">
        <h4>Heading</h4>
        <ul data-role="listview">
            <li><a href="#">Item 1</a></li>
            <li><a href="#">Item 2</a></li>
            <li><a href="#">Item 3</a></li>
        </ul>
    </div>
</body>
</html>
```

## Output

Let's carry out the following steps to see how the above code works:

- Save the above html code as **collapsible\_expanded.html** file in your server root folder.
- Open this HTML file as [http://localhost/collapsible\\_expanded.html](http://localhost/collapsible_expanded.html) and the following output will be displayed.

## Heading

- [Item 1](#)
- [Item 2](#)
- [Item 3](#)

## Mini Sized Collapsible

---

### Description

Mini sized can be used whenever you need a compact version of collapsible. This can be achieved by using *data-mini* and setting it to *true*.

### Example

Following example demonstrates the use of mini sized collapsible in jQuery Mobile.

```
<!DOCTYPE html>
<head>
    <title>Mini sized collapsible</title>
    <meta name="viewport" content="width=device-width, initial-scale=1">
    <link rel="stylesheet"
    href="https://code.jquery.com/mobile/1.4.5/jquery.mobile-1.4.5.min.css">
    <script src="https://code.jquery.com/jquery-1.11.3.min.js"></script>
    <script src="https://code.jquery.com/mobile/1.4.5/jquery.mobile-
    1.4.5.min.js"></script>
</head>
<body>
    <div data-role="collapsible" data-mini="true">
        <h4>Heading</h4>
        <ul data-role="listview">
            <li><a href="#">Item 1</a></li>
            <li><a href="#">Item 2</a></li>
            <li><a href="#">Item 3</a></li>
        </ul>
    </div>
</body>
</html>
```

## Output

Let's carry out the following steps to see how the above code works:

- Save the above html code as **collapsible\_mini\_sized.html** file in your server root folder.
- Open this HTML file as [http://localhost/collapsible\\_mini\\_sized.html](http://localhost/collapsible_mini_sized.html) and the following output will be displayed.

### Heading

- [Item 1](#)
- [Item 2](#)
- [Item 3](#)

## Collapsible Icons

---

### Description

The collapsible heading's default icons can be overriden using ***data-collapsed-icon*** and ***data-expanded-icon*** attributes.

### Example

Following example demonstrates the use of collapsible icons in jQuery Mobile.

```
<!DOCTYPE html>
<head>
    <title>Collapsible Icons</title>
    <meta name="viewport" content="width=device-width, initial-scale=1">
    <link rel="stylesheet"
    href="https://code.jquery.com/mobile/1.4.5/jquery.mobile-1.4.5.min.css">
    <script src="https://code.jquery.com/jquery-1.11.3.min.js"></script>
    <script src="https://code.jquery.com/mobile/1.4.5/jquery.mobile-
    1.4.5.min.js"></script>
</head>
<body>
    <div data-role="collapsible" data-collapsed-icon="carat-d" data-expanded-
    icon="carat-u">
        <h4>Heading</h4>
        <ul data-role="listview" data-inset="false">
```

```

<li>Read-only item 1</li>
<li>Read-only item 2</li>
<li>Read-only item 3</li>
</ul>
</div>
</body>
</html>

```

## Output

Let's carry out the following steps to see how the above code works:

- Save the above html code as **collapsible\_icons.html** file in your server root folder.
- Open this HTML file as [http://localhost/collapsible\\_icons.html](http://localhost/collapsible_icons.html) and the following output will be displayed.

### Heading

- Read-only item 1
- Read-only item 2
- Read-only item 3

## Collapsible Icons Positioning

---

### Description

The collapsible heading's default icon positioning can be overridden using **data-iconpos** attribute. This attribute can be set with *left*, *right*, *top*, or *bottom* value.

### Example

Following example demonstrates the use of collapsible icons positioning in jQuery Mobile.

```

<!DOCTYPE html>
<head>
    <title>Collapsible Icons Positioning</title>
    <meta name="viewport" content="width=device-width, initial-scale=1">
    <link rel="stylesheet"
        href="https://code.jquery.com/mobile/1.4.5/jquery.mobile-1.4.5.min.css">

```

```

<script src="https://code.jquery.com/jquery-1.11.3.min.js"></script>
<script src="https://code.jquery.com/mobile/1.4.5/jquery.mobile-
1.4.5.min.js"></script>
</head>
<body>
    <div data-role="collapsible" data-iconpos="right">
        <h3>Click here to view your message</h3>
        <p>Hi!! Have a nice day!!</p>
    </div>
</body>
</html>

```

## Output

Let's carry out the following steps to see how the above code works:

- Save the above html code as **collapsible\_icons\_positioning.html** file in your server root folder.
- Open this HTML file as [http://localhost/collapsible\\_icons\\_positioning.html](http://localhost/collapsible_icons_positioning.html) and the following output will be displayed.

**Click here to view your message**

Hi!! Have a nice day!!

## Collapsible Legend

---

### Description

You can display the form elements in collapsible using a *fieldset* and *legend*.

### Example

Following example demonstrates the use of collapsible forms using fieldset in jQuery Mobile.

```

<!DOCTYPE html>
<head>
```

```
<title>Collapsible Legend</title>
<meta name="viewport" content="width=device-width, initial-scale=1">
<link rel="stylesheet"
href="https://code.jquery.com/mobile/1.4.5/jquery.mobile-1.4.5.min.css">
<script src="https://code.jquery.com/jquery-1.11.3.min.js"></script>
<script src="https://code.jquery.com/mobile/1.4.5/jquery.mobile-
1.4.5.min.js"></script>
</head>
<body>
<form>
  <fieldset data-role="collapsible">
    <legend>Legend</legend>
    <label for="text-basic">Text input:</label>
    <input type="text" name="text-basic" id="text-basic" value="" placeholder="Input your text">
    <fieldset data-role="controlgroup">
      <legend>Select your favourite sport</legend>
      <input type="checkbox" name="checkbox-1" id="checkbox-1" checked="">
      <label for="checkbox-1">Baseball</label>
      <input type="checkbox" name="checkbox-2" id="checkbox-2">
      <label for="checkbox-2">Cricket</label>
      <input type="checkbox" name="checkbox-3" id="checkbox-3">
      <label for="checkbox-3">Football</label>
      <input type="checkbox" name="checkbox-4" id="checkbox-4">
      <label for="checkbox-4">Hockey</label>
    </fieldset>
  </fieldset>
</form>
</body>
</html>
```

## Output

Let's carry out the following steps to see how the above code works:

- Save the above html code as **collapsible\_legend.html** file in your server root folder.
- Open this HTML file as [http://localhost/collapsible\\_legend.html](http://localhost/collapsible_legend.html) and the following output will be displayed.

The screenshot shows a 'Legend' section with a title bar. Inside, there is a text input field with the placeholder 'Input your text'. Below it is a group of four checkboxes labeled 'Baseball', 'Cricket', 'Football', and 'Hockey'. The 'Baseball' checkbox is checked, while the others are unchecked.

## Non-inset Collapsible

---

### Description

To remove the inset appearance and to display the collapsible to its full width without corner styling, you can add the **`data-inset="false"`** attribute.

### Example

Following example demonstrates the use of non-inset collapsible in jQuery Mobile.

```
<!DOCTYPE html>
<head>
    <title>Non-inset Collapsible</title>
    <meta name="viewport" content="width=device-width, initial-scale=1">
    <link rel="stylesheet"
    href="https://code.jquery.com/mobile/1.4.5/jquery.mobile-1.4.5.min.css">
    <script src="https://code.jquery.com/jquery-1.11.3.min.js"></script>
    <script src="https://code.jquery.com/mobile/1.4.5/jquery.mobile-
    1.4.5.min.js"></script>
</head>
<body>
    <div data-role="collapsible" data-inset="false">
        <h3>Heading</h3>
        <p>Hi!! Have a nice day!!</p>
    </div>
</body>
```

```
</html>
```

## Output

Let's carry out the following steps to see how the above code works:

- Save the above html code as **collapsible\_non\_inset.html** file in your server root folder.
- Open this HTML file as [http://localhost/collapsible\\_non\\_inset.html](http://localhost/collapsible_non_inset.html) and the following output will be displayed.



**Heading**

Hi!! Have a nice day!!

## Collapsible Set

### Description

You can apply a series of individual collapsibles and it allows you to open multiple rows at once. The border of the collapsible can be customized by providing custom CSS.

### Example

Following example demonstrates the use of collapsible set in jQuery Mobile.

```
<!DOCTYPE html>
<head>
    <title>Collapsible Set</title>
    <meta name="viewport" content="width=device-width, initial-scale=1">
    <link rel="stylesheet"
    href="https://code.jquery.com/mobile/1.4.5/jquery.mobile-1.4.5.min.css">
    <script src="https://code.jquery.com/jquery-1.11.3.min.js"></script>
    <script src="https://code.jquery.com/mobile/1.4.5/jquery.mobile-
    1.4.5.min.js"></script>
</head>
<body>
    <div data-role="collapsible">
```

```

<h3>Audi</h3>
<ul data-role="listview">
    <li><a href="#">Audi A4</a></li>
    <li><a href="#">Audi Q3</a></li>
    <li><a href="#">Audi R8</a></li>
</ul>
</div>
<div data-role="collapsible">
    <h3>Mahindra</h3>
    <ul data-role="listview">
        <li><a href="#">XUV 500</a></li>
        <li><a href="#">Scorpio</a></li>
        <li><a href="#">Xylo</a></li>
    </ul>
</div>
<div data-role="collapsible">
    <h3>Hyundai</h3>
    <ul data-role="listview">
        <li><a href="#">Verna</a></li>
        <li><a href="#">i20</a></li>
        <li><a href="#">Elantra</a></li>
    </ul>
</div>
</body>
</html>

```

## Output

Let's carry out the following steps to see how the above code works:

- Save the above html code as **collapsible\_set.html** file in your server root folder.
- Open this HTML file as [http://localhost/collapsible\\_set.html](http://localhost/collapsible_set.html) and the following output will be displayed.

## Audi

- [Audi A4](#)
- [Audi Q3](#)
- [Audi R8](#)

## Mahindra

- [XUV 500](#)
- [Scorpio](#)
- [Xylo](#)

## Hyundai

- [Verna](#)
- [i20](#)
- [Elantra](#)

## Collapsible Pre-rendered Markup

### Description

To improve the load time of your page, you can provide a pre-rendered markup for collapsible by adding **data-enhanced** attribute and setting it to *true*.

### Example

Following example demonstrates the use of collapsible *pre-rendered* markup in jQuery Mobile.

```
<!DOCTYPE html>
<head>
    <title>Collapsible Pre-rendered markup</title>
    <meta name="viewport" content="width=device-width, initial-scale=1">
    <link rel="stylesheet"
    href="https://code.jquery.com/mobile/1.4.5/jquery.mobile-1.4.5.min.css">
    <script src="https://code.jquery.com/jquery-1.11.3.min.js"></script>
    <script src="https://code.jquery.com/mobile/1.4.5/jquery.mobile-
    1.4.5.min.js"></script>
</head>
<body>
    <div data-role="collapsible" data-enhanced="true" class="ui-collapsible ui-
    collapsible-inset ui-corner-all ui-collapsible-collapsed">
```

```

<h4 class="ui-collapsible-heading ui-collapsible-heading-collapsed">
    <a href="#" class="ui-collapsible-heading-toggle ui-btn ui-btn-icon-left ui-icon-plus">Heading
        <div class="ui-collapsible-heading-status"> to expand contents click here</div>
    </a>
</h4>
<div class="ui-collapsible-content ui-collapsible-content-collapsed" aria-hidden="true">
    <p>Hi!!!... Have a nice day!!!..</p>
</div>
</div>
</body>
</html>

```

## Output

Let's carry out the following steps to see how the above code works:

- Save the above html code as **collapsible\_pre\_rendered\_markup.html** file in your server root folder.
- Open this HTML file as [http://localhost/collapsible\\_pre\\_rendered\\_markup.html](http://localhost/collapsible_pre_rendered_markup.html) and the following output will be displayed.

[Heading](#)  
[to expand contents click here](#)

Hi!!!... Have a nice day!!!..

## Dynamic Collapsible

---

### Description

Collapsible can be added to a collapsible set dynamically.

### Example

Following example demonstrates the use of dynamic collapsible in jQuery Mobile.

```

<!DOCTYPE html>
<head>
    <title>Dynamic collapsible</title>
    <meta name="viewport" content="width=device-width, initial-scale=1">
    <link rel="stylesheet"
    href="https://code.jquery.com/mobile/1.4.5/jquery.mobile-1.4.5.min.css">
    <script src="https://code.jquery.com/jquery-1.11.3.min.js"></script>
    <script src="https://code.jquery.com/mobile/1.4.5/jquery.mobile-
1.4.5.min.js"></script>
</head>
<body>
    <button type="button" data-icon="plus" data-iconpos="right" data-mini="true"
data-inline="true" id="add_collapsible">Add collapsible</button>
    <button type="button" data-icon="carat-d" data-iconpos="right" data-
mini="true" data-inline="true" id="expand_last">Expand last
collapsible</button>
    <button type="button" data-icon="carat-u" data-iconpos="right" data-
mini="true" data-inline="true" id="collapse_last">Collapse last</button>
    <div data-role="collapsibleset" data-content-theme="a" data-iconpos="right"
id="set">
        <div data-role="collapsible" id="set1" data-collapsed="true">
            <h3>Heading 1</h3>
            <p>Hi!! This is collapsible content.</p>
        </div>
    </div>
</script>
$( document ).on( "pagecreate", function() {
    var nextId = 1;
    $("#add_collapsible").click(function() {
        nextId++;
        var content = "<div data-role='collapsible' id='set" + nextId +
"'><h3>Heading " + nextId + "</h3><p>Hi!! This is collapsible
content.</p></div>";
        $( "#set" ).append( content ).collapsibleset( "refresh" );
    });
    $("#expand_last").click(function() {
        $("#set").children(":last").collapsible( "expand" );
    });
    $("#collapse_last").click(function() {
        $("#set").children(":last").collapsible( "collapse" );
    });
});

```

```

    });
});
</script>
</body>
</html>

```

## Output

Let's carry out the following steps to see how the above code works:

- Save the above html code as **dynamic\_collapsible.html** file in your server root folder.
- Open this HTML file as [http://localhost/dynamic\\_collapsible.html](http://localhost/dynamic_collapsible.html) and the following output will be displayed.

[Add collapsible](#) [Expand last collapsible](#) [Collapse last](#)

### Heading 1

Hi!! This is collapsible content.

## Collapsible Set

---

### Description

In jQuery Mobile an accordion can be created by putting a series of separate collapsible into set.

Following table lists different types of collapsible set used in jQuery Mobile:

| Sr. No. | Collapsible set types & Description  |
|---------|--|
| 1       | <b>Markup</b><br>The collapsibles set can be created with the same markup as basic collapsible. Adding <b><i>data-role="collapsiblset"</i></b> attribute will make it a collapsible set. |
| 2       | <b>Inset vs. full width</b><br>To create a full width collapsible without styling the corners, add <i>data-inset</i> attribute and set to <i>false</i> .                                 |
| 3       | <b>Mini</b>  |

|   |   |
|---|---|
|   | Whenever you want a compact version of a set of collapsible, add <i>data-mini</i> attribute and set it to <i>true</i> .   |
| 4 | <b><u>Icons</u></b><br>To override the default heading icons, add <i>data-collapsed-icon</i> and <i>data-expanded-icon</i> attributes at <i>collapsibleset</i> level or on whichever of its collapsibles. |
| 5 | <b><u>Icon position</u></b><br>To override the default icon position, use the <i>data-iconpos</i> attribute at <i>collapsibleset</i> level or on whichever of its collapsibles.                           |
| 6 | <b><u>Corners</u></b><br>To get an inset collapsible set without rounded corner styling just add <i>data-corners</i> attribute and set to <i>false</i> .  |
| 7 | <b><u>Theme</u></b><br>The color of each collapsible header can be set by adding a <i>data-theme</i> attribute to the collapsible set.  |

## Collapsible Markup

---

### Description

The collapsibles set can be created with the same markup as basic collapsible. Adding ***data-role="collapsibleset"*** attribute will make it a collapsible set and behave like a accordion.

### Example

Following example demonstrates the use of collapsible set in jQuery Mobile.

```
<!DOCTYPE html>
<head>
  <title>Collapsible Markup</title>
  <meta name="viewport" content="width=device-width, initial-scale=1">
  <link rel="stylesheet"
  href="https://code.jquery.com/mobile/1.4.5/jquery.mobile-1.4.5.min.css">
  <script src="https://code.jquery.com/jquery-1.11.3.min.js"></script>
  <script src="https://code.jquery.com/mobile/1.4.5/jquery.mobile-
  1.4.5.min.js"></script>
</head>
<body>
  <div data-role="collapsibleset" data-theme="a" data-content-theme="a">
    <div data-role="collapsible">
```

```
<h3>Heading 1</h3>
<p>Hi!! This is collapsible content for heading 1</p>
</div>
<div data-role="collapsible">
    <h3>Heading 2</h3>
    <p>Hi!! This is collapsible content for heading 2</p>
</div>
<div data-role="collapsible">
    <h3>Heading 3</h3>
    <p>Hi!! This is collapsible content for heading 3</p>
</div>
</div>
</body>
</html>
```

## Output

Let's carry out the following steps to see how the above code works:

- Save the above html code as **collapsible\_markup.html** file in your server root folder.
- Open this HTML file as [http://localhost/collapsible\\_markup.html](http://localhost/collapsible_markup.html) and the following output will be displayed.

## Heading 1

Hi!! This is collapsible content for heading 1

## Heading 2

Hi!! This is collapsible content for heading 2

## Heading 3

Hi!! This is collapsible content for heading 3

## **Collapsible Inset vs. Full Width**

---

### **Description**

To create a full width collapsible without styling the corners, add *data-inset* attribute and set to *false*. This displays the collapsibles set as an expandable [listview](#).

## **jQuery Mobile - Listview**

---

### **Description**

The Listview is used to display a list of items. ***data-role="listview"*** attribute is included in the container to display the list in a vertical scrollable list.

Following table demonstrates the types of listview in detail.

| Sr. No. | Types & Description   |
|---------|---|
| 1       | <b><u>Read-only, unordered</u></b><br>Include <i>data-role="listview"</i> attribute to create a simple unordered list.                      |
| 2       | <b><u>Read-only, ordered</u></b><br>Create ordered lists(ol) by including <i>data-role="listview"</i> attribute in it.                      |
| 3       | <b><u>Linked</u></b><br>When a hyperlink is defined in the list, then it is styled as a button.   |
| 4       | <b><u>Inset</u></b><br>Apply the inset look by including the attribute <i>data-inset="true"</i> .   |
| 5       | <b><u>Filter</u></b><br>The list can be filtered by adding the <i>data-filter="true"</i> attribute.   |
| 6       | <b><u>Filter reveal</u></b><br>Add the <i>data-filter-reveal="true"</i> attribute to hide the list item.                                    |
| 7       | <b><u>List dividers</u></b><br>Add <i>data-role="list-divider"</i> attribute to any list item to divide into a group of list items.         |
| 8       | <b><u>Autodividers</u></b><br><i>data-autodividers="true"</i> attribute is included to generate the dividers automatically to any listview. |
| 9       | <b><u>Count bubbles</u></b>   |

|    |  |
|----|--|
|    | <i>ui-li-count</i> class define in element will help to include a count indicator in the list item at the right side.  |
| 10 | <b><u>Icons: Standard</u></b><br>Add attribute <i>data-icon</i> to set icon for a particular list item.  |
| 11 | <b><u>Icons: 16x16</u></b><br>Include class of <i>ui-li-icon</i> in the image element to add image of 16 X 16 pixels in the list item.                               |
| 12 | <b><u>Thumbnails</u></b><br>In list item, the thumbnails is included on the left side.   |
| 13 | <b><u>Split buttons</u></b><br>Include a second link inside the <i>/i</i> to create split list item.   |
| 14 | <b><u>Formatted content</u></b><br>Use heading and paragraph tags to add content in the proper hierarchy format.   |
| 15 | <b><u>Theme</u></b><br><i>data-theme</i> attribute is added to include theme for the listview or to a particular list item.  |
| 16 | <b><u>Forms</u></b><br>In listview, the form element can be inserted for a group presentation.   |
| 17 | <b><u>Collapsible listview</u></b><br><i>data-role="collapsible"</i> attribute is wrapped in the container to make the listview collapsible.                         |
| 18 | <b><u>Grouped collapsible with listviews</u></b><br>A grouped listview can be created inside the collapsible set using attribute <i>data-role="collapsibleset"</i> . |
| 19 | <b><u>Full width collapsible listview</u></b><br><i>data-inset="false"</i> attribute is added to make full width collapsible set.                                    |

## Listview Unordered Widget

---

### Description

Include *data-role="listview"* attribute to create a simple unordered list.

### Example

Following example demonstrates the use of unordered list in the jQuery Mobile.

```

<!DOCTYPE html>
<html>
<head>
    <meta name="viewport" content="width=device-width, initial-scale=1">
    <link rel="stylesheet"
    href="https://code.jquery.com/mobile/1.4.5/jquery.mobile-1.4.5.min.css">
    <script src="https://code.jquery.com/jquery-1.11.3.min.js"></script>
    <script src="https://code.jquery.com/mobile/1.4.5/jquery.mobile-
1.4.5.min.js"></script>
</head>
<body>
<h2>Listview Unordered Example</h2>
<ul data-role="listview">
    <li>Chennai</li>
    <li>Bangalore</li>
    <li>Pune</li>
    <li>Mumbai</li>
    <li>Delhi</li>
</ul>
</body>
</html>

```

## Output

Let's carry out the following steps to see how the above code works:

- Save the above html code as **listview\_unordered.html** file in your server root folder.
- Open this HTML file as [http://localhost/listview\\_unordered.html](http://localhost/listview_unordered.html) and the following output will be displayed.

### Listview Unordered Example

- Chennai
- Bangalore
- Pune
- Mumbai
- Delhi

## Listview Ordered

---

### Description

Create an ordered lists(ol) by including ***data-role="listview"*** attribute in it.

### Example

Following example demonstrates the use of ordered list in the jQuery Mobile.

```
<!DOCTYPE html>
<html>
<head>
    <meta name="viewport" content="width=device-width, initial-scale=1">
    <link rel="stylesheet"
    href="https://code.jquery.com/mobile/1.4.5/jquery.mobile-1.4.5.min.css">
    <script src="https://code.jquery.com/jquery-1.11.3.min.js"></script>
    <script src="https://code.jquery.com/mobile/1.4.5/jquery.mobile-
1.4.5.min.js"></script>
</head>
<body>
<h2>Listview Ordered Example</h2>
<ol data-role="listview">
    <li>Chennai</li>
    <li>Bangalore</li>
    <li>Pune</li>
    <li>Mumbai</li>
    <li>Delhi</li>
</ol>
</body>
</html>
```

### Output

Let's carry out the following steps to see how the above code works:

- Save the above html code as **listview\_ordered.html** file in your server root folder.
- Open this HTML file as [http://localhost/listview\\_ordered.html](http://localhost/listview_ordered.html) and the following output will be displayed.

## Listview Ordered Example

1. Chennai
2. Bangalore
3. Pune
4. Mumbai
5. Delhi

## Listview Linked

---

### Description

When a hyperlink is defined in the list, then it is styled as a button.

### Example

Following example demonstrates the use of linked list in the jQuery Mobile.

```
<!DOCTYPE html>
<html>
<head>
    <meta name="viewport" content="width=device-width, initial-scale=1">
    <link rel="stylesheet"
    href="https://code.jquery.com/mobile/1.4.5/jquery.mobile-1.4.5.min.css">
    <script src="https://code.jquery.com/jquery-1.11.3.min.js"></script>
    <script src="https://code.jquery.com/mobile/1.4.5/jquery.mobile-
    1.4.5.min.js"></script>
</head>
<body>
<h2>Listview Linked Example</h2>
<ul data-role="listview">
    <li><a href="#">Chennai</a></li>
    <li><a href="#">Bangalore</a></li>
    <li><a href="#">Pune</a></li>
    <li><a href="#">Mumbai</a></li>
    <li><a href="#">Delhi</a></li>
```

```

</ul>
</body>
</html>
```

## Output

Let's carry out the following steps to see how the above code works:

- Save the above html code as **listview\_linked.html** file in your server root folder.
- Open this HTML file as [http://localhost/listview\\_linked.html](http://localhost/listview_linked.html) and the following output will be displayed.

## Listview Linked Example

- [Chennai](#)
- [Bangalore](#)
- [Pune](#)
- [Mumbai](#)
- [Delhi](#)

## Listview Inset

---

### Description

Apply inset look by including the attribute ***data-inset="true"***. It helps in mixing a listview with other content on a page.

### Example

Following example demonstrates the use of listview inset in the jQuery Mobile.

```

<!DOCTYPE html>
<html>
<head>
    <meta name="viewport" content="width=device-width, initial-scale=1">
    <link rel="stylesheet"
    href="https://code.jquery.com/mobile/1.4.5/jquery.mobile-1.4.5.min.css">
    <script src="https://code.jquery.com/jquery-1.11.3.min.js"></script>
    <script src="https://code.jquery.com/mobile/1.4.5/jquery.mobile-
    1.4.5.min.js"></script>
```

```

</head>
<body>
<h2>Listview Inset Example</h2>
<ul data-role="listview" data-inset="true">
    <li><a href="">Chennai</a></li>
    <li><a href="">Bangalore</a></li>
    <li><a href="">Pune</a></li>
    <li><a href="">Mumbai</a></li>
    <li><a href="">Delhi</a></li>
</ul>
</body>
</html>

```

## Output

Let's carry out the following steps to see how the above code works:

- Save the above html code as **listview\_inset.html** file in your server root folder.
- Open this HTML file as [http://localhost/listview\\_inset.html](http://localhost/listview_inset.html) and the following output will be displayed.

## Listview Inset Example

- [Chennai](#)
- [Bangalore](#)
- [Pune](#)
- [Mumbai](#)
- [Delhi](#)

## Listview Filter

---

### Description

The list can be filtered by adding the ***data-filter="true"*** attribute. A search box is included above the list and behavior to filter out the list items is added. By default, the theme is inherited from the parents for search box.

## Example

Following example demonstrates the use of filter list in the jQuery Mobile.

```
<!DOCTYPE html>
<html>
<head>
    <meta name="viewport" content="width=device-width, initial-scale=1">
    <link rel="stylesheet"
    href="https://code.jquery.com/mobile/1.4.5/jquery.mobile-1.4.5.min.css">
    <script src="https://code.jquery.com/jquery-1.11.3.min.js"></script>
    <script src="https://code.jquery.com/mobile/1.4.5/jquery.mobile-
1.4.5.min.js"></script>
</head>
<body>
<h2>Listview Filter Example</h2>
<ul data-role="listview" data-inset="true" data-filter="true" data-filter-
placeholder="Search">
    <li><a href="#">Chennai</a></li>
    <li><a href="#">Bangalore</a></li>
    <li><a href="#">Pune</a></li>
    <li><a href="#">Mumbai</a></li>
    <li><a href="#">Delhi</a></li>
    <li><a href="#">Dehradun</a></li>
    <li><a href="#">Punjab</a></li>
</ul>
</body>
</html>
```

## Output

Let's carry out the following steps to see how the above code works:

- Save the above html code as **listview\_filter.html** file in your server root folder.

- Open this HTML file as [http://localhost/listview\\_filter.html](http://localhost/listview_filter.html) and the following output will be displayed.

## Listview Filter Example

- [Chennai](#)
- [Bangalore](#)
- [Pune](#)
- [Mumbai](#)
- [Delhi](#)
- [Dehradun](#)
- [Punjab](#)

## Listview Filter Reveal

---

### Description

When the search box is empty then it will auto hide the complete list item. Add the ***data-filter-reveal="true"*** attribute to the hide the list item.

### Example

Following example demonstrates the use of filter list reveal in the jQuery Mobile.

```
<!DOCTYPE html>
<html>
<head>
    <meta name="viewport" content="width=device-width, initial-scale=1">
    <link rel="stylesheet"
    href="https://code.jquery.com/mobile/1.4.5/jquery.mobile-1.4.5.min.css">
    <script src="https://code.jquery.com/jquery-1.11.3.min.js"></script>
    <script src="https://code.jquery.com/mobile/1.4.5/jquery.mobile-
    1.4.5.min.js"></script>
</head>
<body>
<h2>Listview Filter Reveal Example</h2>
<ul data-role="listview" data-inset="true" data-filter="true" data-filter-
reveal="Search" data-filter-placeholder="Search">
    <li><a href="#">Chennai</a></li>
    <li><a href="#">Bangalore</a></li>
    <li><a href="#">Pune</a></li>
    <li><a href="#">Mumbai</a></li>
```

```

<li><a href="#">Delhi</a></li>
<li><a href="#">Dehradhun</a></li>
<li><a href="#">Punjab</a></li>
</ul>
</body>
</html>

```

## Output

Let's carry out the following steps to see how the above code works:

- Save the above html code as **listview\_filter\_reveal.html** file in your server root folder.
- Open this HTML file as [http://localhost/listview\\_filter\\_reveal.html](http://localhost/listview_filter_reveal.html) and the following output will be displayed.

## Listview Filter Reveal Example

- [Chennai](#)
- [Bangalore](#)
- [Pune](#)
- [Mumbai](#)
- [Delhi](#)
- [Dehradhun](#)
- [Punjab](#)

## Listview Dividers

---

### Description

Add ***data-role="list-divider"*** attribute to any list item to divide into a group of list items. ***data-divider-theme*** attribute can be used to include the theme to the list item.

### Example

Following example demonstrates the use of list divider in the jQuery Mobile.

```

<!DOCTYPE html>
<html>
<head>

```

```
<meta name="viewport" content="width=device-width, initial-scale=1">
<link rel="stylesheet"
href="https://code.jquery.com/mobile/1.4.5/jquery.mobile-1.4.5.min.css">
<script src="https://code.jquery.com/jquery-1.11.3.min.js"></script>
<script src="https://code.jquery.com/mobile/1.4.5/jquery.mobile-
1.4.5.min.js"></script>
</head>
<body>
<h2>List Dividers Example</h2>
<ul data-role="listview" data-inset="true" data-divider-theme="b">
<li data-role="list-divider">Karnataka</li>
<li><a href="#">Bangalore</a></li>
<li><a href="#">Mangalore</a></li>
<li><a href="#">Hubli</a></li>
<li data-role="list-divider">Maharashtra</li>
<li><a href="#">Pune</a></li>
<li><a href="#">Mumbai</a></li>
</ul>
</body>
</html>
```

## Output

Let's carry out the following steps to see how the above code works:

- Save the above html code as **listview\_list\_dividers.html** file in your server root folder.
- Open this HTML file as [http://localhost/listview\\_list\\_dividers.html](http://localhost/listview_list_dividers.html) and the following output will be displayed.

## List Dividers Example

- Karnataka
- [Bangalore](#)
- [Mangalore](#)
- [Hubli](#)
- Maharashtra
- [Pune](#)
- [Mumbai](#)

## Listview Autodividers

---

### Description

`data-autodividers="true"` attribute is included to generate the dividers automatically to any listview.

### Example

Following example demonstrates the use of list auto divider in the jQuery Mobile.

```
<!DOCTYPE html>
<html>
<head>
    <meta name="viewport" content="width=device-width, initial-scale=1">
    <link rel="stylesheet"
    href="https://code.jquery.com/mobile/1.4.5/jquery.mobile-1.4.5.min.css">
    <script src="https://code.jquery.com/jquery-1.11.3.min.js"></script>
    <script src="https://code.jquery.com/mobile/1.4.5/jquery.mobile-
    1.4.5.min.js"></script>
</head>
<body>
<h2>List Autodividers Example</h2>
<ul data-role="listview" data-autodividers="true" data-filter="true" data-
inset="true">
    <li><a href="#">Dharwad</a></li>
    <li><a href="#">Davangere</a></li>
    <li><a href="#">Mangalore</a></li>
    <li><a href="#">Mumbai</a></li>
    <li><a href="#">Pune</a></li>
```

```

<li><a href="#">Punjab</a></li>
</ul>
</body>
</html>

```

## Output

Let's carry out the following steps to see how the above code works:

- Save the above html code as **listview\_autodividers.html** file in your server root folder.
- Open this HTML file as [http://localhost/listview\\_autodividers.html](http://localhost/listview_autodividers.html) and the following output will be displayed.

## List Autodividers Example

- [Dharwad](#)
- [Davangere](#)
- [Mangalore](#)
- [Mumbai](#)
- [Pune](#)
- [Punjab](#)

## Listview Count Bubbles

---

### Description

**ui-li-count** class define in element will help to include a count indicator in the list item on the right side. Add **data-count-theme** attribute to set the theme for count bubbles in the list.

### Example

Following example demonstrates the use of count bubble list in the jQuery Mobile.

```

<!DOCTYPE html>
<html>
<head>
    <meta name="viewport" content="width=device-width, initial-scale=1">
    <link rel="stylesheet"
    href="https://code.jquery.com/mobile/1.4.5/jquery.mobile-1.4.5.min.css">
    <script src="https://code.jquery.com/jquery-1.11.3.min.js"></script>

```

```

<script src="https://code.jquery.com/mobile/1.4.5/jquery.mobile-
1.4.5.min.js"></script>

</head>

<body>

<h2>Count Bubble List Example</h2>

<ul data-role="listview" data-count-theme="b" data-inset="true">
    <li><a href="#">Inbox<span class="ui-li-count">100</span></a></li>
    <li><a href="#">Starred<span class="ui-li-count">5</span></a></li>
    <li><a href="#">Important<span class="ui-li-count">2</span></a></li>
    <li><a href="#">Chats<span class="ui-li-count">6</span></a></li>
    <li><a href="#">Sent Mail<span class="ui-li-count">20</span></a></li>
    <li><a href="#">Drafts<span class="ui-li-count">12</span></a></li>
</ul>
</body>
</html>

```

## Output

Let's carry out the following steps to see how the above code works:

- Save the above html code as **listview\_count\_bubbles.html** file in your server root folder.
- Open this HTML file as [http://localhost/listview\\_count\\_bubbles.html](http://localhost/listview_count_bubbles.html) and the following output will be displayed.

## Count Bubble List Example

- [Inbox100](#)
- [Starred5](#)
- [Important2](#)
- [Chats6](#)
- [Sent Mail20](#)
- [Drafts12](#)

## Listview Standard Icons

---

### Description

Add attribute **`data-icon`** to set icon for a particular list item, you can set different types of icons by including the name of any standard icon.

## Example

Following example demonstrates the use of Icon standard in the list in the jQuery Mobile.

```
<!DOCTYPE html>
<html>
<head>
    <meta name="viewport" content="width=device-width, initial-scale=1">
    <link rel="stylesheet"
    href="https://code.jquery.com/mobile/1.4.5/jquery.mobile-1.4.5.min.css">
    <script src="https://code.jquery.com/jquery-1.11.3.min.js"></script>
    <script src="https://code.jquery.com/mobile/1.4.5/jquery.mobile-
1.4.5.min.js"></script>
</head>
<body>
<h2>Icon Standard List Example</h2>
<ul data-role="listview" data-inset="true">
    <li data-icon="custom"><a href="#">Inbox</a></li>
    <li data-icon="gear"><a href="#">Setting</a></li>
    <li data-icon="info"><a href="#">Important</a></li>
    <li data-icon="delete"><a href="#">Chats</a></li>
</ul>
</body>
</html>
```

## Output

Let's carry out the following steps to see how the above code works:

- Save the above html code as **`listview_icons_standard.html`** file in your server root folder.
- Open this HTML file as `http://localhost/listview_icons_standard.html` and the following output will be displayed.

## Icon Standard List Example

- [Inbox](#)
- [Setting](#)
- [Important](#)
- [Chats](#)

## Listview Icons: 16x16

### Description

Include class of ***ui-li-icon*** in the image element to add image of 16 X 16 pixels in the list item.

### Example

Following example demonstrates the use of Icon 16 X 16 in the list in the jQuery Mobile.

```
<!DOCTYPE html>
<html>
<head>
    <meta name="viewport" content="width=device-width, initial-scale=1">
    <link rel="stylesheet"
    href="https://code.jquery.com/mobile/1.4.5/jquery.mobile-1.4.5.min.css">
    <script src="https://code.jquery.com/jquery-1.11.3.min.js"></script>
    <script src="https://code.jquery.com/mobile/1.4.5/jquery.mobile-
    1.4.5.min.js"></script>
</head>
<body>
    <h2>Icon 16X16 List Example</h2>
    <ul data-role="listview" data-inset="true">
        <li><a href="#">Apple</a></li>
        <li><a href="#">Banana</a></li>
        <li><a href="#">Mango</a></li>
        <li><a href="#">Strawberry</a></li>
    </ul>
</body>
</html>
```

```
<li><a href="#">Grapes</a></li>
</ul>
</body>
</html>
```

## Output

Let's carry out the following steps to see how the above code works:

- Save the above html code as **listview\_icons\_16.html** file in your server root folder.
- Open this HTML file as [http://localhost/listview\\_icons\\_16.html](http://localhost/listview_icons_16.html) and the following output will be displayed.

### Icon 16X16 List Example



[Apple](#)

- [Banana](#)
- [Mango](#)
- [Strawberry](#)



## Listview Thumbnails

---

### Description

In list item, the thumbnails are included on the left side. Insert the image as the first child element inside the list item.

### Example

Following example demonstrates the use of thumbnails in the list in the jQuery Mobile.

```
<!DOCTYPE html>
<html>
<head>
    <meta name="viewport" content="width=device-width, initial-scale=1">
    <link rel="stylesheet"
    href="https://code.jquery.com/mobile/1.4.5/jquery.mobile-1.4.5.min.css">
    <script src="https://code.jquery.com/jquery-1.11.3.min.js"></script>
    <script src="https://code.jquery.com/mobile/1.4.5/jquery.mobile-
    1.4.5.min.js"></script>
</head>
<body>
    <h2>Listview Thumbnail Example</h2>
    <ul data-role="listview" data-inset="true">
        <li><a href="#">
            
```

```
<h2>Facebook</h2>
<p>Mark Zukerburg</p></a>
</li>
<li><a href="#">
    
    <h2>Twitter</h2>
    <p>Jack Dorsey</p></a>
</li>
<li><a href="#">
    
    <h2>Google+</h2>
    <p>Google</p></a>
</li>
</ul>
</body>
</html>
```

## Output

Let's carry out the following steps to see how the above code works:

- Save the above html code as **listview\_thumbnail.html** file in your server root folder.
- Open this HTML file as [http://localhost/listview\\_thumbnail.html](http://localhost/listview_thumbnail.html) and the following output will be displayed.

## Listview Thumbnail Example



[Facebook](#)

[Mark Zukerburg](#)



[Twitter](#)

[Jack Dorsey](#)



[Google+](#)

[Google](#)

## Listview Split buttons

---

### Description

Include a second link inside the ***li*** to create split list item. ***data-split-icon*** attribute is used to adjust the split button icon and ***data-split-theme*** attribute is used to include theme for the split button.

### Example

Following example demonstrates the use of split button in the list in the jQuery Mobile.

```
<!DOCTYPE html>
<html>
<head>
    <meta name="viewport" content="width=device-width, initial-scale=1">
    <link rel="stylesheet"
    href="https://code.jquery.com/mobile/1.4.5/jquery.mobile-1.4.5.min.css">
    <script src="https://code.jquery.com/jquery-1.11.3.min.js"></script>
```

```

<script src="https://code.jquery.com/mobile/1.4.5/jquery.mobile-
1.4.5.min.js"></script>

</head>

<body>

<h2>Listview Split Button Example</h2>

<ul data-role="listview" data-split-icon="gear" data-split-theme="a" data-
inset="true">

<li><a href="#">
    
    <h2>Facebook</h2>
    <p>Mark Zukerburg</p></a>
    <a href="#purchase" data-rel="popup" data-position-to="window" data-
transition="pop">Login</a>
</li>

<li><a href="#">
    <h2>Twitter</h2>
    <p>Jack Dorsey</p></a>
    <a href="#purchase" data-rel="popup" data-position-to="window" data-
transition="pop">Login</a>
</li>

<li><a href="#">
    <h2>Google+</h2>
    <p>Google</p></a>
    <a href="#purchase" data-rel="popup" data-position-to="window" data-
transition="pop">Login</a>
</li>
</ul>

<div data-role="popup" id="purchase" data-theme="a" data-overlay-theme="b"
class="ui-content">
    <h3>Social Media Sites</h3>
    <p>Login in your mobile device</p>
    <a href="index.html" data-rel="back" class="ui-shadow ui-btn ui-btn-b ui-
icon-check ui-btn-icon-left ui-btn-inline ui-mini">LOGIN</a>
    <a href="index.html" data-rel="back" class="ui-shadow ui-btn ui-btn-inline
ui-mini">Cancel</a>
</div>
</body>
</html>

```

## Output

Let's carry out the following steps to see how the above code works:

- Save the above html code as **listview\_split\_buttons.html** file in your server root folder.
- Open this HTML file as [http://localhost/listview\\_split\\_buttons.html](http://localhost/listview_split_buttons.html) and the following output is displayed.

**Listview Split Button Example**



[Facebook](#)

[Mark Zuckerberg](#)

[Login](#)



[Twitter](#)

[Jack Dorsey](#)

[Login](#)



[Google+](#)

[Google](#)

[Login](#)

Social Media Sites

Login in your mobile device

[LOGIN](#) [Cancel](#)

## Listview Formatted Content

---

### Description

Use heading and paragraph tags to add content in the proper hierarchy format. Include class **ui-li-aside** to add supplemental information on the right side of the list item.

## Example

Following example demonstrates the use of Thumbnails in the list in the jQuery Mobile.

```
<!DOCTYPE html>
<html>
<head>
    <meta name="viewport" content="width=device-width, initial-scale=1">
    <link rel="stylesheet"
    href="https://code.jquery.com/mobile/1.4.5/jquery.mobile-1.4.5.min.css">
    <script src="https://code.jquery.com/jquery-1.11.3.min.js"></script>
    <script src="https://code.jquery.com/mobile/1.4.5/jquery.mobile-
1.4.5.min.js"></script>
</head>
<body>
<h2>Listview Formatted Content Example</h2>
<ul data-role="listview" data-inset="true">
    <li data-role="list-divider">Wednesday, January 13, 2016 <span class="ui-li-
count">2</span></li>
    <li><a href="#">
        <h2>John</h2>
        <p><strong>Team Meeting</strong></p>
        <p>Lorem Ipsum is simply dummy text of the printing and typesetting
industry.</p></a>
        <p class="ui-li-aside"><strong>12:48</strong>PM</p>
    </li>
    <li><a href="#">
        <h2>Albert</h2>
        <p><strong>Updation of work</strong></p>
        <p>Lorem Ipsum is simply dummy text of the printing and typesetting
industry.</p></a>
        <p class="ui-li-aside"><strong>4:01</strong>PM</p>
    </li>
    <li data-role="list-divider">Thursday, January 14, 2016 <span class="ui-li-
count">2</span></li>
    <li><a href="#"><h2>Joy</h2>
        <p><strong>New Project Information</strong></p>
        <p>Lorem Ipsum is simply dummy text of the printing and typesetting
industry.</p></a>
        <p class="ui-li-aside"><strong>09:25</strong>PM</p>
    </li>
</ul>
```

```

</li>
</ul>
</body>
</html>
```

## Output

Let's carry out the following steps to see how the above code works:

- Save the above html code as **listview\_formatted\_content.html** file in your server root folder.
- Open this HTML file as [http://localhost/listview\\_formatted\\_content.html](http://localhost/listview_formatted_content.html) and the following output will be displayed.

## Listview Formatted Content Example

- Wednesday, January 13, 2016 2

- **John**

**Team Meeting**

[Lorem Ipsum is simply dummy text of the printing and typesetting industry.](#)

12:48PM

- **Albert**

**Updation of work**

[Lorem Ipsum is simply dummy text of the printing and typesetting industry.](#)

4:01PM

- Thursday, January 14, 2016 2

- **Joy**

**New Project Information**

[Lorem Ipsum is simply dummy text of the printing and typesetting industry.](#)

09:25PM

## Listview Theme

---

### Description

**data-theme** attribute is added to include the theme for the listview or to the particular list item.

- **data-divider-theme** attribute sets the theme for the list divider.
- **data-count-theme** attribute is used to set the theme for the count bubbles.
- **data-split-theme** attribute sets the theme for the split button.

### Example

Following example demonstrates the use of theme in the list in the jQuery Mobile.

```
<!DOCTYPE html>
<html>
<head>
    <meta name="viewport" content="width=device-width, initial-scale=1">
    <link rel="stylesheet"
    href="https://code.jquery.com/mobile/1.4.5/jquery.mobile-1.4.5.min.css">
    <script src="https://code.jquery.com/jquery-1.11.3.min.js"></script>
    <script src="https://code.jquery.com/mobile/1.4.5/jquery.mobile-
1.4.5.min.js"></script>
</head>
<body>
<h2>Listview Theme Example</h2>
<h3>Count bubble theme</h3>
<ul data-role="listview" data-inset="true" data-theme="b" data-divider-
theme="a" data-count-theme="a">
    <li data-role="list-divider">Mail Box</li>
    <li><a href="#">Inbox <span class="ui-li-count">100</span></a></li>
    <li><a href="#">Chats<span class="ui-li-count">6</span></a></li>
    <li><a href="#">Sent Mail<span class="ui-li-count">20</span></a></li>
</ul><br/>

<h3>Split Button theme</h3>
<ul data-role="listview" data-split-icon="plus" data-theme="a" data-split-
theme="b" data-inset="true">
    <li><a href="">Facebook</a>
        <a href="#purchase" data-rel="popup" data-position-to="window" data-
transition="pop">Login</a>

```

```

</li>
<li><a href="">Twitter</a>
    <a href="#purchase" data-theme="a" data-rel="popup" data-position-
to="window" data-transition="pop">Login</a>
</li>
</ul>
</body>
</html>

```

## Output

Let's carry out the following steps to see how the above code works:

- Save the above html code as **listview\_theme.html** file in your server root folder.
- Open this HTML file as [http://localhost/listview\\_theme.html](http://localhost/listview_theme.html) and the following output will be displayed.

## Listview Theme Example

### Count bubble theme

- Mail Box
- [Inbox 100](#)
- [Chats6](#)
- [Sent Mail120](#)

### Split Button theme

- [Facebook Login](#)
- [Twitter Login](#)

## Listview Forms

---

### Description

In listview, the form element can be inserted for a group presentation.

## Example

Following example demonstrates the use of Thumbnails in the list in the jQuery Mobile.

```
<!DOCTYPE html>
<html>
<head>
    <meta name="viewport" content="width=device-width, initial-scale=1">
    <link rel="stylesheet"
    href="https://code.jquery.com/mobile/1.4.5/jquery.mobile-1.4.5.min.css">
    <script src="https://code.jquery.com/jquery-1.11.3.min.js"></script>
    <script src="https://code.jquery.com/mobile/1.4.5/jquery.mobile-
1.4.5.min.js"></script>
</head>
<body>
<h2>Listview Form Example</h2>
<form>
<ul data-role="listview" data-inset="true">
    <li class="ui-field-contain">
        <label for="name">Text</label>
        <input type="text" id="name" data-clear-btn="true">
    </li>
    <li class="ui-field-contain">
        <label for="textarea">Textarea</label>
        <textarea cols="40" rows="8" id="textarea"></textarea>
    </li>
    <li class="ui-field-contain">
        <label for="flip">Flip switch</label>
        <select id="flip" data-role="slider">
            <option value="off">Off</option>
            <option value="on">On</option>
        </select>
    </li>
    <li class="ui-field-contain">
        <label for="slider">Slider</label>
        <input type="range" id="slider" value="0" min="0" max="100" data-
highlight="true">
    </li>
    <li class="ui-field-contain">
```

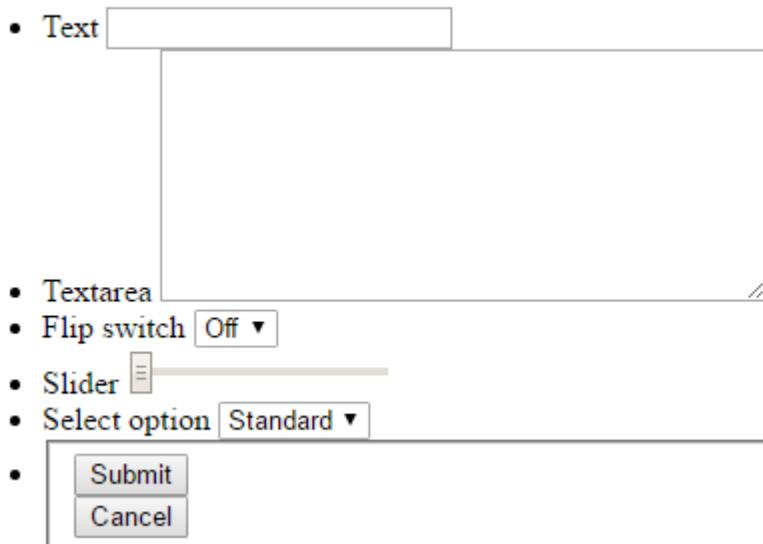
```
<label for="select-choice" class="select">Select option</label>
<select name="select-choice" id="select-choice-1">
    <option>Standard</option>
    <option>Premium</option>
    <option>Deluxe</option>
</select>
</li>
<li class="ui-body ui-body-b">
    <fieldset class="ui-grid-a">
        <div class="ui-block-a"><button type="submit" class="ui-btn ui-corner-all ui-btn-a">Submit</button></div>
        <div class="ui-block-b"><button type="Cancel" class="ui-btn ui-corner-all ui-btn-a">Cancel</button></div>
    </fieldset>
</li>
</ul>
</form>
</body>
</html>
```

## Output

Let's carry out the following steps to see how the above code works:

- Save the above html code as **listview\_forms.html** file in your server root folder.
- Open this HTML file as [http://localhost/listview\\_forms.html](http://localhost/listview_forms.html) and the following output will be displayed.

## Listview Form Example



## Listview Collapsible

### Description

**data-role="collapsible"** attribute is wrapped in the container to make the listview collapsible.

### Example

Following example demonstrates the use of collapsible list in the jQuery Mobile.

```
<!DOCTYPE html>
<html>
<head>
    <meta name="viewport" content="width=device-width, initial-scale=1">
    <link rel="stylesheet"
    href="https://code.jquery.com/mobile/1.4.5/jquery.mobile-1.4.5.min.css">
    <script src="https://code.jquery.com/jquery-1.11.3.min.js"></script>
    <script src="https://code.jquery.com/mobile/1.4.5/jquery.mobile-
    1.4.5.min.js"></script>
</head>
<body>
<div data-role="collapsible" data-theme="b" data-content-theme="b">
    <h2>Listview Collapsible Example</h2>

```

```

<ul data-role="listview" data-filter="true">
    <li><a href="#">Chennai</a></li>
    <li><a href="#">Belgaum</a></li>
    <li><a href="#">Bangalore</a></li>
    <li><a href="#">Mumbai</a></li>
    <li><a href="#">Pune</a></li>
    <li><a href="#">Delhi</a></li>
    <li><a href="#">Punjab</a></li>
</ul>
</div>
</body>
</html>

```

## Output

Let's carry out the following steps to see how the above code works:

- Save the above html code as **listview\_collapsible.html** file in your server root folder.
- Open this HTML file as [http://localhost/listview\\_collapsible.html](http://localhost/listview_collapsible.html) and the following output will be displayed.

## Listview Collapsible Example

- [Chennai](#)
- [Belgaum](#)
- [Bangalore](#)
- [Mumbai](#)
- [Pune](#)
- [Delhi](#)
- [Punjab](#)

## Grouped Collapsible With Listviews

---

### Description

A grouped listview can be created inside the collapsible set using attribute **data-role="collapsibleset"**. Only one list item can be opened at a time.

## Example

Following example demonstrates the use of grouped collapsible with listviews in the jQuery Mobile.

```
<!DOCTYPE html>
<html>
<head>
    <meta name="viewport" content="width=device-width, initial-scale=1">
    <link rel="stylesheet"
    href="https://code.jquery.com/mobile/1.4.5/jquery.mobile-1.4.5.min.css">
    <script src="https://code.jquery.com/jquery-1.11.3.min.js"></script>
    <script src="https://code.jquery.com/mobile/1.4.5/jquery.mobile-
1.4.5.min.js"></script>
</head>
<body>
<div data-role="collapsibleset" data-theme="a" data-content-theme="b">
    <div data-role="collapsible">
        <h2>Thumbnail</h2>
        <ul data-role="listview" data-split-icon="gear" data-split-theme="a"
data-inset="true">
            <li><a href="#">
                
                <h2>Facebook</h2>
                <p>Mark Zukerburg</p></a>
                <a href="#purchase" data-rel="popup" data-position-to="window"
data-transition="pop">Login</a>
            </li>
            <li><a href="#">
                <h2>Twitter</h2>
                <p>Jack Dorsey</p></a>
                <a href="#purchase" data-rel="popup" data-position-to="window"
data-transition="pop">Login</a>
            </li>
            <li><a href="#">
                <h2>Google+</h2>
                <p>Google</p></a>
                <a href="#purchase" data-rel="popup" data-position-to="window"
data-transition="pop">Login</a>
            </li>
        </ul>
    </div>
</div>
```

```

</ul>

<div data-role="popup" id="purchase" data-theme="a" data-overlay-theme="b" class="ui-content" >
    <h3>Social Media Sites</h3>
    Login in your mobile device
    <a href="index.html" data-rel="back" class="ui-shadow ui-btn ui-btn-b ui-icon-check ui-btn-icon-left ui-btn-inline ui-mini">Log In</a>
    <a href="index.html" data-rel="back" class="ui-shadow ui-btn ui-btn-inline ui-mini">Cancel</a>
</div>
</div>
<div data-role="collapsible">
    <h2>Formatted Content</h2>
    <ul data-role="listview" data-inset="true">
        <li data-role="list-divider">Wednesday, January 13, 2016 <span class="ui-li-count">2</span></li>
        <li><a href="#">
            <h2>John</h2>
            <p><strong>Team Meeting</strong></p>
            <p>Lorem Ipsum is simply dummy text of the printing and typesetting industry.</p></a>
            <p class="ui-li-aside"><strong>12:48</strong>PM</p>
        </li>
        <li><a href="#">
            <h2>Albert</h2>
            <p><strong>Updation of work</strong></p>
            <p>Lorem Ipsum is simply dummy text of the printing and typesetting industry.</p></a>
            <p class="ui-li-aside"><strong>4:01</strong>PM</p>
        </li>
        <li data-role="list-divider">Thursday, January 14, 2016 <span class="ui-li-count">2</span></li>
        <li><a href="#">
            <h2>Joy</h2>
            <p><strong>New Project Information</strong></p>
            <p>Lorem Ipsum is simply dummy text of the printing and typesetting industry.</p></a>
            <p class="ui-li-aside"><strong>09:25</strong>PM</p>
        </li>
    </ul>
</div>

```

```
</ul>
</div>
<div data-role="collapsible">
    <h2> Filter</h2>
    <ul data-role="listview" data-inset="true" data-filter="true" data-
filter-placeholder="Search">
        <li><a href="#">Chennai</a></li>
        <li><a href="#">Belgaum</a></li>
        <li><a href="#">Bangalore</a></li>
        <li><a href="#">Pune </a></li>
        <li><a href="#">Mumbai</a></li>
        <li><a href="#">Delhi</a></li>
    </ul>
</div>
</div>
</body>
</html>
```

## Output

Let's carry out the following steps to see how the above code works:

- Save the above html code as **listview\_grouped\_collapsible.html** file in your server root folder.
- Open this HTML file as [http://localhost/listview\\_grouped\\_collapsible.html](http://localhost/listview_grouped_collapsible.html) and the following output will be displayed.

## Thumbnail



### [Facebook](#)

[Mark Zuckerberg](#)

[Login](#)



### [Twitter](#)

[Jack Dorsey](#)

[Login](#)



## [Google+](#)

[Google](#)

[Login](#)



## Social Media Sites

◆◆◆◆ Login in your mobile device ◆◆◆◆ [Log In](#) ◆◆◆◆ [Cancel](#)

## Formatted Content

- Wednesday, January 13, 2016 2

- [John](#)

[Team Meeting](#)

[Lorem Ipsum](#) is simply dummy text of the printing and typesetting industry.

12:48PM

- [Albert](#)

[Updation of work](#)

[Lorem Ipsum is simply dummy text of the printing and typesetting industry.](#)

**4:01PM**

- Thursday, January 14, 2016 2

- [Joy](#)

[New Project Information](#)

[Lorem Ipsum is simply dummy text of the printing and typesetting industry.](#)

**09:25PM**

### Filter

- [Chennai](#)
- [Belgaum](#)
- [Bangalore](#)
- [Pune](#)
- [Mumbai](#)
- [Delhi](#)

## Full Width Collapsible Listview

---

### Description

***data-inset="false"*** attribute is added to make full width collapsible set.

### Example

Following example demonstrates the use of full width collapsible listview in the jQuery Mobile.

```
<!DOCTYPE html>
<html>
<head>
    <meta name="viewport" content="width=device-width, initial-scale=1">
    <link rel="stylesheet"
    href="https://code.jquery.com/mobile/1.4.5/jquery.mobile-1.4.5.min.css">
    <script src="https://code.jquery.com/jquery-1.11.3.min.js"></script>
    <script src="https://code.jquery.com/mobile/1.4.5/jquery.mobile-
    1.4.5.min.js"></script>
</head>
<body>
```

```

<div data-role="collapsibleset" data-theme="a" data-inset="false">
  <div data-role="collapsible">
    <h2>Thumbnail</h2>
    <ul data-role="listview" data-split-icon="gear" data-split-theme="a"
      data-inset="true">
      <li><a href="#">
        
        <h2>Facebook</h2>
        <p>Mark Zukerburg</p></a>
        <a href="#purchase" data-rel="popup" data-position-to="window"
          data-transition="pop">Login</a>
      </li>
      <li><a href="#">
        <h2>Twitter</h2>
        <p>Jack Dorsey</p></a>
        <a href="#purchase" data-rel="popup" data-position-to="window"
          data-transition="pop">Login</a>
      </li>
      <li><a href="#">
        <h2>Google+</h2>
        <p>Google</p></a>
        <a href="#purchase" data-rel="popup" data-position-to="window"
          data-transition="pop">Login</a>
      </li>
    </ul>
    <div data-role="popup" id="purchase" data-theme="a" data-overlay-
      theme="b" class="ui-content" >
      <h3>Social Media Sites</h3>
      Login in your mobile device
      <a href="index.html" data-rel="back" class="ui-shadow ui-btn ui-btn-b
        ui-icon-check ui-btn-icon-left ui-btn-inline ui-mini">Log In</a>
      <a href="index.html" data-rel="back" class="ui-shadow ui-btn ui-btn-
        inline ui-mini">Cancel</a>
    </div>
  </div>
  <div data-role="collapsible">
    <h2>Formatted Content</h2>
    <ul data-role="listview" data-inset="true">

```

```

<li data-role="list-divider">Wednesday, January 13, 2016 <span
class="ui-li-count">2</span></li>
<li><a href="#">
    <h2>John</h2>
    <p><strong>Team Meeting</strong></p>
    <p>Lorem Ipsum is simply dummy text of the printing and typesetting
industry.</p></a>
    <p class="ui-li-aside"><strong>12:48</strong>PM</p>
</li>
<li><a href="#">
    <h2>Albert</h2>
    <p><strong>Updation of work</strong></p>
    <p>Lorem Ipsum is simply dummy text of the printing and typesetting
industry.</p></a>
    <p class="ui-li-aside"><strong>4:01</strong>PM</p>
</li>
<li data-role="list-divider">Thursday, January 14, 2016 <span
class="ui-li-count">2</span></li>
<li><a href="#"><h2>Joy</h2>
    <p><strong>New Project Information</strong></p>
    <p>Lorem Ipsum is simply dummy text of the printing and typesetting
industry.</p></a>
    <p class="ui-li-aside"><strong>09:25</strong>PM</p>
</li>
</ul>
</div>
<div data-role="collapsible">
    <h2> Filter</h2>
    <ul data-role="listview" data-inset="true" data-filter="true" data-
filter-placeholder="Search">
        <li><a href="#">Chennai</a></li>
        <li><a href="#">Belgaum</a></li>
        <li><a href="#">Bangalore</a></li>
        <li><a href="#">Pune </a></li>
        <li><a href="#">Mumbai</a></li>
        <li><a href="#">Delhi</a></li>
    </ul>
</div>
</div>

```

```
</body>  
</html>
```

## Output

Let's carry out the following steps to see how the above code works:

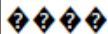
- Save the above html code as **listview\_width\_collapsible.html** file in your server root folder.
- Open this HTML file as [http://localhost/listview\\_width\\_collapsible.html](http://localhost/listview_width_collapsible.html) and the following output will be displayed.

**Thumbnail**

-  [Facebook](#)  
[Mark Zukerburg](#)  
[Login](#)
-  [Twitter](#)  
[Jack Dorsey](#)  
[Login](#)
-  [Google+](#)

[Google](#)

[Login](#)



## Social Media Sites

?? ?? Login in your mobile device ?? ?? [Log In](#) ?? ?? [Cancel](#)

## Formatted Content

- Wednesday, January 13, 2016 2

- **[John](#)**

[Team Meeting](#)

[Lorem Ipsum is simply dummy text of the printing and typesetting industry.](#)

12:48PM

- **[Albert](#)**

[Updation of work](#)

[Lorem Ipsum is simply dummy text of the printing and typesetting industry.](#)

4:01PM

- Thursday, January 14, 2016 2

- **[Joy](#)**

[New Project Information](#)

[Lorem Ipsum is simply dummy text of the printing and typesetting industry.](#)

09:25PM

## Filter

- [Chennai](#)
- [Belgaum](#)
- [Bangalore](#)
- [Pune](#)
- [Mumbai](#)
- [Delhi](#)

## Example

Following example demonstrates the use of collapsible set without corner styles in jQuery Mobile.

```
<!DOCTYPE html>
<head>
    <title>Collapsible Inset vs. full width</title>
    <meta name="viewport" content="width=device-width, initial-scale=1">
    <link rel="stylesheet"
    href="https://code.jquery.com/mobile/1.4.5/jquery.mobile-1.4.5.min.css">
    <script src="https://code.jquery.com/jquery-1.11.3.min.js"></script>
    <script src="https://code.jquery.com/mobile/1.4.5/jquery.mobile-
1.4.5.min.js"></script>
</head>
<body>
    <div data-role="collapsibleset" data-inset="false">
        <div data-role="collapsible">
            <h3>Audi</h3>
            <ul data-role="listview" data-inset="false">
                <li>Audi A3</li>
                <li>Audi Q3</li>
                <li>Audi A4</li>
                <li>Audi A3 cabriolet</li>
            </ul>
        </div>
        <div data-role="collapsible">
            <h3>Hyundai</h3>
            <ul data-role="listview" data-inset="false">
                <li>Verna</li>
                <li>Elantra</li>
                <li>Santa Fe</li>
                <li>Creta</li>
            </ul>
        </div>
        <div data-role="collapsible">
            <h3>Toyota</h3>
            <ul data-role="listview" data-inset="false">
                <li>Corolla Altis</li>
```

```

<li>Fortuner</li>
<li>Land Cruiser Prado</li>
<li>Land Cruiser 200</li>
</ul>
</div>
</div>
</body>
</html>

```

## Output

Let's carry out the following steps to see how the above code works:

- Save the above html code as **collapsible\_inset\_fullwidth.html** file in your server root folder.
- Open this HTML file as [http://localhost/collapsible\\_inset\\_fullwidth.html](http://localhost/collapsible_inset_fullwidth.html) and the following output will be displayed.

**Audi**

- Audi A3
- Audi Q3
- Audi A4
- Audi A3 cabriolet

**Hyundai**

- Verna
- Elantra
- Santa Fe
- Creta

**Toyota**

- Corolla Altis
- Fortuner
- Land Cruiser Prado
- Land Cruiser 200

## Mini Collapsible Set

---

### Description

Whenever you want a compact version of a set of collapsible, add ***data-mini*** attribute and set it to *true*.

## Example

Following example demonstrates the use of compact version collapsible set in jQuery Mobile.

```
<!DOCTYPE html>
<head>
    <title>Mini collapsible set </title>
    <meta name="viewport" content="width=device-width, initial-scale=1">
    <link rel="stylesheet"
    href="https://code.jquery.com/mobile/1.4.5/jquery.mobile-1.4.5.min.css">
    <script src="https://code.jquery.com/jquery-1.11.3.min.js"></script>
    <script src="https://code.jquery.com/mobile/1.4.5/jquery.mobile-
1.4.5.min.js"></script>
</head>
<body>
    <div data-role="collapsibleset" data-theme="a" data-content-theme="a" data-
mini="true">
        <div data-role="collapsible">
            <h3>Click here</h3>
            <p>Hi!! This is mini collapsible</p>
        </div>
        <div data-role="collapsible">
            <h3>Click here</h3>
            <p>Hi!! This is another mini collapsible</p>
        </div>
        <div data-role="collapsible">
            <h3>Click here</h3>
            <p>Hi!! This is yet another mini collapsible</p>
        </div>
    </div>
</body>
</html>
```

## Output

Let's carry out the following steps to see how the above code works:

- Save the above html code as **collapsible\_mini.html** file in your server root folder.

- Open this HTML file as [http://localhost/collapsible\\_mini.html](http://localhost/collapsible_mini.html) and the following output will be displayed.

## Click here

Hi!! This is mini collapsible

## Click here

Hi!! This is another mini collapsible

## Click here

Hi!! This is yet another mini collapsible

## Collapsible Set Icon

---

### Description

To override the default heading icons, add ***data-collapsed-icon*** and ***data-expanded-icon*** attributes at ***collapsiblset*** level or on whichever of its collapsibles.

### Example

Following example demonstrates the use of collapsible set icon in jQuery Mobile.

```
<!DOCTYPE html>
<head>
    <title>Collapsible set icon</title>
    <meta name="viewport" content="width=device-width, initial-scale=1">
    <link rel="stylesheet"
    href="https://code.jquery.com/mobile/1.4.5/jquery.mobile-1.4.5.min.css">
    <script src="https://code.jquery.com/jquery-1.11.3.min.js"></script>
    <script src="https://code.jquery.com/mobile/1.4.5/jquery.mobile-
    1.4.5.min.js"></script>
</head>
<body>
    <div data-role="collapsiblset" data-theme="a" data-content-theme="a" data-
    collapsed-icon="carat-d" data-expanded-icon="carat-u">
        <div data-role="collapsible">
            <h3>Heading</h3>
```

```

<p>Hi!! This is collapsible content</p>
</div>
<div data-role="collapsible">
    <h3>Heading</h3>
    <p>Hi!! This is collapsible content</p>
</div>
<div data-role="collapsible" data-collapsed-icon="plus" data-expanded-icon="minus">
    <h3>Heading</h3>
    <p>Hi!! This is collapsible content</p>
</div>
</div>
</body>
</html>

```

## Output

Let's carry out the following steps to see how the above code works:

- Save the above html code as **collapsible\_set\_icon.html file** in your server root folder.
- Open this HTML file as [http://localhost/collapsible\\_set\\_icon.html](http://localhost/collapsible_set_icon.html) and the following output will be displayed.

### **Heading**

Hi!! This is collapsible content

### **Heading**

Hi!! This is collapsible content

### **Heading**

Hi!! This is collapsible content

## Collapsible Set Icon Positioning

---

### Description

To override the default icon position, use the ***data-iconpos*** attribute at ***collapsibleset*** level or on whichever of its collapsibles.

### Example

Following example demonstrates the use of collapsible set icon in jQuery Mobile.

```
<!DOCTYPE html>
<head>
<title>Collapsible set icon</title>
<meta name="viewport" content="width=device-width, initial-scale=1">
<link rel="stylesheet"
href="https://code.jquery.com/mobile/1.4.5/jquery.mobile-1.4.5.min.css">
<script src="https://code.jquery.com/jquery-1.11.3.min.js"></script>
<script src="https://code.jquery.com/mobile/1.4.5/jquery.mobile-
1.4.5.min.js"></script>
</head>
<body>
<div data-role="collapsibleset" data-theme="a" data-content-theme="a" data-
iconpos="right">
<div data-role="collapsible">
<h3>Right</h3>
<p>Hi!! This is collapsible content</p>
</div>
<div data-role="collapsible" data-iconpos="left">
<h3>Left</h3>
<p>Hi!! This is collapsible content</p>
</div>
<div data-role="collapsible" data-iconpos="bottom">
<h3>Bottom</h3>
<p>Hi!! This is collapsible content</p>
</div>
<div data-role="collapsible" data-iconpos="top">
<h3>Top</h3>
<p>Hi!! This is collapsible content</p>
</div>
</div>
```

```
</body>
</html>
```

## Output

Let's carry out the following steps to see how the above code works:

- Save the above html code as **collapsible\_set\_icon\_positioning.html** file in your server root folder.
- Open this HTML file as [http://localhost/collapsible\\_set\\_icon\\_positioning.html](http://localhost/collapsible_set_icon_positioning.html) and the following output will be displayed.

### Right

Hi!! This is collapsible content

### Left

Hi!! This is collapsible content

### Bottom

Hi!! This is collapsible content

### Top

Hi!! This is collapsible content

## Collapsible Set Without Rounded Corners

### Description

To get an inset collapsible set without rounded corner styling just add **data-corners** attribute and set to *false*.

### Example

Following example demonstrates the use of collapsible set without rounded corner in jQuery Mobile.

```
<!DOCTYPE html>
```

```

<head>
    <title>Collapsible set without rounded corners</title>
    <meta name="viewport" content="width=device-width, initial-scale=1">
    <link rel="stylesheet"
    href="https://code.jquery.com/mobile/1.4.5/jquery.mobile-1.4.5.min.css">
    <script src="https://code.jquery.com/jquery-1.11.3.min.js"></script>
    <script src="https://code.jquery.com/mobile/1.4.5/jquery.mobile-
1.4.5.min.js"></script>
</head>
<body>
    <div data-role="collapsibleset" data-corners="false" data-theme="a" data-
content-theme="a">
        <div data-role="collapsible">
            <h3>Heading 1</h3>
            <p>This is collapsible content</p>
        </div>
        <div data-role="collapsible">
            <h3>Heading 2</h3>
            <p>This is collapsible content</p>
        </div>
        <div data-role="collapsible">
            <h3>Heading 3</h3>
            <p>This is collapsible content</p>
        </div>
    </div>
</body>
</html>

```

## Output

Let's carry out the following steps to see how the above code works:

- Save the above html code as **collapsible\_set\_corners.html** file in your server root folder.
- Open this HTML file as [http://localhost/collapsible\\_set\\_corners.html](http://localhost/collapsible_set_corners.html) and the following output will be displayed.

## Heading 1

This is collapsible content

## Heading 2

This is collapsible content

## Heading 3

This is collapsible content

## **Collapsible Set Theme**

---

### Description

The color of each collapsible header can be set by adding a data-theme attribute to the collapsible set.

To specify the content's color, you can add data-content-theme attribute. You can differently style individual sections in a group by adding data-theme and data-content-theme attributes to specific collapsibles.

### Example

Following example demonstrates the use of collapsible theme in jQuery Mobile.

```
<!DOCTYPE html>
<head>
<title>Collapsible set theme</title>
<meta name="viewport" content="width=device-width, initial-scale=1">
<link rel="stylesheet"
href="https://code.jquery.com/mobile/1.4.5/jquery.mobile-1.4.5.min.css">
<script src="https://code.jquery.com/jquery-1.11.3.min.js"></script>
<script src="https://code.jquery.com/mobile/1.4.5/jquery.mobile-
1.4.5.min.js"></script>
</head>
<body>
<div data-role="collapsibleset" data-theme="b" data-content-theme="b">
    <div data-role="collapsible">
        <h3>Heading</h3>
```

```

<p>This is collapsible content</p>
</div>
<div data-role="collapsible">
    <h3>Heading</h3>
    <p>This is collapsible content</p>
</div>
</div>

<p>You can differently style individual sections in a group by adding
<i>data-theme</i> and <i>data-content-theme</i> attributes to specific
collapsibles.</p>

<div data-role="collapsibleset" data-theme="b" data-content-theme="b">
    <div data-role="collapsible">
        <h3>Heading</h3>
        <p>This is collapsible content</p>
    </div>
    <div data-role="collapsible" data-theme="a" data-content-theme="a">
        <h3>Heading</h3>
        <p>This is collapsible content</p>
    </div>
    <div data-role="collapsible">
        <h3>Heading</h3>
        <p>This is collapsible content</p>
    </div>
</div>
</body>
</html>

```

## Output

Let's carry out the following steps to see how the above code works:

- Save the above html code as **collapsible\_set\_theme.html** file in your server root folder.
- Open this HTML file as [http://localhost/collapsible\\_set\\_theme.html](http://localhost/collapsible_set_theme.html) and the following output will be displayed.

## Heading

This is collapsible content

## Heading

This is collapsible content

You can differently style individual sections in a group by adding *data-theme* and *data-content-theme* attributes to specific collapsibles.

## Heading

This is collapsible content

## Heading

This is collapsible content

## Heading

This is collapsible content

## Nested Collapsible

---

### Description

Collapsible can be nested inside other collapsible content.

### Example

Following example demonstrates the use of nested collapsible in jQuery Mobile.

```
<!DOCTYPE html>
<head>
    <title>Collapsible nested</title>
    <meta name="viewport" content="width=device-width, initial-scale=1">
    <link rel="stylesheet"
    href="https://code.jquery.com/mobile/1.4.5/jquery.mobile-1.4.5.min.css">
    <script src="https://code.jquery.com/jquery-1.11.3.min.js"></script>
    <script src="https://code.jquery.com/mobile/1.4.5/jquery.mobile-
    1.4.5.min.js"></script>
</head>
```

```

<body>
  <div data-role="collapsible">
    <h1>Click me</h1>
    <p>This is collapsible content.</p>
    <div data-role="collapsible">
      <h1>Click me</h1>
      <p>This is nested collapsible content.</p>
    </div>
  </div>
</body>
</html>

```

## Output

Let's carry out the following steps to see how the above code works:

- Save the above html code as **collapsible\_nested.html** file in your server root folder.
- Open this HTML file as [http://localhost/collapsible\\_nested.html](http://localhost/collapsible_nested.html) and the following output will be displayed.

**Click me**

This is collapsible content.

**Click me**

This is nested collapsible content.

## jQuery Mobile – Controlgroup Widget

### Description

Controlgroups provide a set of buttons to specify a single block that looks like a navigation component.

Following table lists some of the controlgroups, supported by the jQuery Mobile.

| Sr. No. | Type & Description  |
|---------|---|
| 1       | <b><u>Controlgroup Vertical</u></b><br>You can display the controlgroup in vertical type.                           |
| 2       | <b><u>Controlgroup Horizontal</u></b><br>You can display the controlgroup in horizontal type.                       |
| 3       | <b><u>Mini sized Controlgroup</u></b><br>You can display the controlgroup in small size.                            |
| 4       | <b><u>Controlgroup Radio button</u></b><br>Display the radio button in the controlgroup.                            |
| 5       | <b><u>Controlgroup Checkbox</u></b><br>Display the checkbox in the controlgroup.                                    |
| 6       | <b><u>Controlgroup Selects</u></b><br>Create the dropdown list in the controlgroup.                                 |
| 7       | <b><u>Controlgroup Textinputs</u></b><br>You can specify the text inputs that allow the users to input data.        |
| 8       | <b><u>Controlgroup Pre-rendered Markup</u></b><br>You can use this for saving the startup time in the controlgroup. |

## Controlgroup Vertical Widget

---

### Description

You can display the controlgroup in a vertical format using the ***data-role="controlgroup"*** attribute.

### Example

Following example describes the use of controlgroup vertical in the jQuery Mobile Framework.

```
<!DOCTYPE html>
<head>
<title>Controlgroup Vertical</title>
<meta name="viewport" content="width=device-width, initial-scale=1">
<link rel="stylesheet"
href="https://code.jquery.com/mobile/1.4.5/jquery.mobile-1.4.5.min.css">
```

```

<script src="https://code.jquery.com/jquery-1.11.3.min.js"></script>
<script src="https://code.jquery.com/mobile/1.4.5/jquery.mobile-
1.4.5.min.js"></script>
</head>
<body>
<div data-role="controlgroup">
    <a href="#" class="ui-btn">Without Icon</a>
    <a href="#" class="ui-btn ui-icon-home ui-btn-icon-left">Left Icon</a>
    <a href="#" class="ui-btn ui-icon-home ui-btn-icon-right">Right Icon</a>
    <a href="#" class="ui-btn ui-icon-home ui-btn-icon-top">Top Icon</a>
    <a href="#" class="ui-btn ui-icon-home ui-btn-icon-bottom">Bottom Icon</a>
    <a href="#" class="ui-btn ui-icon-home ui-btn-icon-notext">Icon only</a>
</div>
</body>
</html>

```

## Output

Let's carry out the following steps to see how the above code works:

- Save the above html code as **jqm\_controlgroup\_vertical.html** file in your server root folder.
- Open this HTML file as [http://localhost/jqm\\_controlgroup\\_vertical.html](http://localhost/jqm_controlgroup_vertical.html) and the following output will be displayed.

[Without Icon](#) [Left Icon](#) [Right Icon](#) [Top Icon](#) [Bottom Icon](#) [Icon only](#)

## Controlgroup Horizontal Widget

### Description

Display the controlgroup in a horizontal format using the **data-role="controlgroup"** and **data-type="horizontal"** attributes.

## Example

Following example describes the use of controlgroup horizontal in the jQuery Mobile Framework.

```
<!DOCTYPE html>
<head>
<title>Controlgroup Horizontal</title>
<meta name="viewport" content="width=device-width, initial-scale=1">
<link rel="stylesheet"
href="https://code.jquery.com/mobile/1.4.5/jquery.mobile-1.4.5.min.css">
<script src="https://code.jquery.com/jquery-1.11.3.min.js"></script>
<script src="https://code.jquery.com/mobile/1.4.5/jquery.mobile-
1.4.5.min.js"></script>
</head>
<body>
<div data-role="controlgroup" data-type="horizontal">
<a href="#" class="ui-btn ui-icon-home ui-btn-icon-left">Left Icon</a>
<a href="#" class="ui-btn ui-icon-home ui-btn-icon-right">Right Icon</a>
</div>
<div data-role="controlgroup" data-type="horizontal">
<a href="#" class="ui-btn ui-icon-home ui-btn-icon-top">Top Icon</a>
<a href="#" class="ui-btn ui-icon-home ui-btn-icon-bottom">Bottom Icon</a>
<a href="#" class="ui-btn ui-icon-home ui-btn-icon-notext">Icon only</a>
</div>
</body>
</html>
```

## Output

Let's carry out the following steps to see how the above code works:

- Save the above html code as **jqm\_controlgroup\_horizontal.html** file in your server root folder.
- Open this HTML file as [http://localhost/jqm\\_controlgroup\\_horizontal.html](http://localhost/jqm_controlgroup_horizontal.html) and the following output will be displayed.

[Left Icon Right Icon](#)  
[Top Icon Bottom Icon Icon only](#)

## Controlgroup Minisized Widget

### Description

You can display the controlgroup in small size using the ***data-mini="true"*** attribute.

### Example

Following example describes the use of controlgroup minisized widget in the jQuery Mobile Framework.

```
<!DOCTYPE html>
<head>
<title>Controlgroup Minisized</title>
<meta name="viewport" content="width=device-width, initial-scale=1">
<link rel="stylesheet"
href="https://code.jquery.com/mobile/1.4.5/jquery.mobile-1.4.5.min.css">
<script src="https://code.jquery.com/jquery-1.11.3.min.js"></script>
<script src="https://code.jquery.com/mobile/1.4.5/jquery.mobile-
1.4.5.min.js"></script>
</head>
<body>
<h2>Minisized Vertical Controlgroup</h2>
<div data-role="controlgroup" data-mini="true">
<a href="#" class="ui-btn ui-icon-home ui-btn-icon-left">Left Icon</a>
<a href="#" class="ui-btn ui-icon-home ui-btn-icon-right">Right Icon</a>
<a href="#" class="ui-btn ui-icon-home ui-btn-icon-top">Top Icon</a>
<a href="#" class="ui-btn ui-icon-home ui-btn-icon-bottom">Bottom Icon</a>
</div><br><br>
<h2>Minisized Horizontal Controlgroup</h2>
<div data-role="controlgroup" data-type="horizontal" data-mini="true">
```

```

<a href="#" class="ui-btn ui-icon-home ui-btn-icon-left">Left Icon</a>
<a href="#" class="ui-btn ui-icon-home ui-btn-icon-right">Right Icon</a>
<a href="#" class="ui-btn ui-icon-home ui-btn-icon-top">Top Icon</a>
<a href="#" class="ui-btn ui-icon-home ui-btn-icon-bottom">Bottom Icon</a>
</div>
</body>
</html>

```

## Output

Let's carry out the following steps to see how the above code works:

- Save the above html code as **jqm\_controlgroup\_minisized.html** file in your server root folder.
- Open this HTML file as [http://localhost/jqm\\_controlgroup\\_minisized.html](http://localhost/jqm_controlgroup_minisized.html) and the following output will be displayed.

## Minimized Vertical Controlgroup

[Left Icon](#) [Right Icon](#) [Top Icon](#) [Bottom Icon](#)

## Minimized Horizontal Controlgroup

[Left Icon](#) [Right Icon](#) [Top Icon](#) [Bottom Icon](#)

## Controlgroup Radio Button

---

### Description

You can display the radio buttons in the controlgroup using **data-role="controlgroup"** attribute.

### Example

Following example describes the use of controlgroup radio button in the jQuery Mobile Framework.

```
<!DOCTYPE html>
```

```
<head>
<title>Controlgroup Radio Button</title>
<meta name="viewport" content="width=device-width, initial-scale=1">
<link rel="stylesheet"
href="https://code.jquery.com/mobile/1.4.5/jquery.mobile-1.4.5.min.css">
<script src="https://code.jquery.com/jquery-1.11.3.min.js"></script>
<script src="https://code.jquery.com/mobile/1.4.5/jquery.mobile-
1.4.5.min.js"></script>
</head>
<body>
<h2>Vertical Radio Button</h2>
<form>
  <fieldset data-role="controlgroup">
    <input type="radio" name="radio1" id="radio11" value="val1"
checked="checked">
      <label for="radio11">India</label>
    <input type="radio" name="radio1" id="radio12" value="val2">
      <label for="radio12">Australia</label>
    <input type="radio" name="radio1" id="radio13" value="val3">
      <label for="radio13">New Zealand</label>
  </fieldset>
</form>
<br><br>
<h2>Minisized Vertical Radio Button</h2>
<form>
  <fieldset data-role="controlgroup" data-mini="true">
    <input type="radio" name="radio1" id="radio14" value="val1"
checked="checked">
      <label for="radio14">India</label>
    <input type="radio" name="radio1" id="radio15" value="val2">
      <label for="radio15">Australia</label>
    <input type="radio" name="radio1" id="radio16" value="val3">
      <label for="radio16">New Zealand</label>
  </fieldset>
</form>
<h2>Horizontal Radio Button</h2>
<form>
  <fieldset data-role="controlgroup" data-type="horizontal">
```

```

<input type="radio" name="radio1" id="radio17" value="val1"
checked="checked">
    <label for="radio17">India</label>
    <input type="radio" name="radio1" id="radio18" value="val2">
    <label for="radio18">Australia</label>
    <input type="radio" name="radio1" id="radio19" value="val3">
    <label for="radio19">New Zealand</label>
</fieldset>
</form>
<br><br>
<h2>Minisized Horizontal Radio Button</h2>
<form>
    <fieldset data-role="controlgroup" data-type="horizontal" data-mini="true">
        <input type="radio" name="radio1" id="radio20" value="val1"
checked="checked">
            <label for="radio20">India</label>
            <input type="radio" name="radio1" id="radio21" value="val2">
            <label for="radio21">Australia</label>
            <input type="radio" name="radio1" id="radio22" value="val3">
            <label for="radio22">New Zealand</label>
    </fieldset>
</form>
</body>
</html>

```

## Output

Let's carry out the following steps to see how the above code works:

- Save the above html code as **jqm\_controlgroup\_radio\_button.html** file in your server root folder.
- Open this HTML file as [http://localhost/jqm\\_controlgroup\\_radio\\_button.html](http://localhost/jqm_controlgroup_radio_button.html) and the following output will be displayed.

## Vertical Radio Button

India  Australia  New Zealand

## Minisized Vertical Radio Button

India  Australia  New Zealand

## Horizontal Radio Button

India  Australia  New Zealand

## Minisized Horizontal Radio Button

India  Australia  New Zealand

## Controlgroup Checkbox

### Description

You can create the checkbox in the controlgroup using the ***type="checkbox"*** attribute.

### Example

Following example describes the use of controlgroup checkbox in the jQuery Mobile Framework.

```
<!DOCTYPE html>
<head>
<title>Controlgroup Checkbox</title>
<meta name="viewport" content="width=device-width, initial-scale=1">
<link rel="stylesheet"
href="https://code.jquery.com/mobile/1.4.5/jquery.mobile-1.4.5.min.css">
<script src="https://code.jquery.com/jquery-1.11.3.min.js"></script>
```

```
<script src="https://code.jquery.com/mobile/1.4.5/jquery.mobile-1.4.5.min.js"></script>
</head>
<body>
<h2>Vertical Checkbox</h2>
<form>
  <fieldset data-role="controlgroup">
    <input type="checkbox" name="checkbox1" id="checkbox11">
    <label for="checkbox11">India</label>
    <input type="checkbox" name="checkbox1" id="checkbox12">
    <label for="checkbox12">Australia</label>
    <input type="checkbox" name="checkbox1" id="checkbox13">
    <label for="checkbox13">New Zealand</label>
  </fieldset>
</form>
<h2>Minimized Vertical Checkbox</h2>
<form>
  <fieldset data-role="controlgroup" data-mini="true">
    <input type="checkbox" name="checkbox1" id="checkbox14">
    <label for="checkbox14">India</label>
    <input type="checkbox" name="checkbox1" id="checkbox15">
    <label for="checkbox15">Australia</label>
    <input type="checkbox" name="checkbox1" id="checkbox16">
    <label for="checkbox16">New Zealand</label>
  </fieldset>
</form>
<h2>Horizontal Checkbox</h2>
<form>
  <fieldset data-role="controlgroup" data-type="horizontal">
    <input type="checkbox" name="checkbox1" id="checkbox17">
    <label for="checkbox17">India</label>
    <input type="checkbox" name="checkbox1" id="checkbox18">
    <label for="checkbox18">Australia</label>
    <input type="checkbox" name="checkbox1" id="checkbox19">
    <label for="checkbox19">New Zealand</label>
  </fieldset>
</form>
```

```
<h2>Minisized Horizontal Checkbox</h2>
<form>
  <fieldset data-role="controlgroup" data-type="horizontal" data-mini="true">
    <input type="checkbox" name="radio1" id="checkbox20">
    <label for="checkbox20">India</label>
    <input type="checkbox" name="radio1" id="checkbox21">
    <label for="checkbox21">Australia</label>
    <input type="checkbox" name="radio1" id="checkbox22">
    <label for="checkbox22">New Zealand</label>
  </fieldset>
</form>
</body>
</html>
```

## Output

Let's carry out the following steps to see how the above code works:

- Save the above html code as **jqm\_controlgroup\_checkbox.html** file in your server root folder.
- Open this HTML file as [http://localhost/jqm\\_controlgroup\\_checkbox.html](http://localhost/jqm_controlgroup_checkbox.html) and the following output will be displayed.

## Vertical Checkbox

India  Australia  New Zealand

## Minisized Vertical Checkbox

India  Australia  New Zealand

## Horizontal Checkbox

India  Australia  New Zealand

## Minisized Horizontal Checkbox

India  Australia  New Zealand

## Controlgroup Selects

### Description

Create the dropdown list in the controlgroup using the ***data-role="controlgroup"*** attribute.

### Example

Following example describes the use of controlgroup selects in the jQuery Mobile Framework.

```
<!DOCTYPE html>
<head>
<title>Controlgroup Selects</title>
<meta name="viewport" content="width=device-width, initial-scale=1">
<link rel="stylesheet"
href="https://code.jquery.com/mobile/1.4.5/jquery.mobile-1.4.5.min.css">
<script src="https://code.jquery.com/jquery-1.11.3.min.js"></script>
<script src="https://code.jquery.com/mobile/1.4.5/jquery.mobile-
1.4.5.min.js"></script>
```

```
</head>
<body>
<h2>Vertical Select</h2>
<form>
  <fieldset data-role="controlgroup">
    <select>
      <option value="#">India</option>
      <option value="#">Australia</option>
      <option value="#">New Zealand</option>
    </select>
  </fieldset>
</form>
<h2>Minisized Vertical Select</h2>
<form>
  <fieldset data-role="controlgroup" data-mini="true">
    <select>
      <option value="#">India</option>
      <option value="#">Australia</option>
      <option value="#">New Zealand</option>
    </select>
  </fieldset>
</form>
<h2>Horizontal Select</h2>
<form>
  <fieldset data-role="controlgroup" data-type="horizontal">
    <select>
      <option value="#">India</option>
      <option value="#">Australia</option>
      <option value="#">New Zealand</option>
    </select>
  </fieldset>
</form>
<h2>Minisized Horizontal Select</h2>
<form>
  <fieldset data-role="controlgroup" data-type="horizontal" data-mini="true">
    <select>
      <option value="#">India</option>
```

```

<option value="#">Australia</option>
<option value="#">New Zealand</option>
</select>
</fieldset>
</form>
</body>
</html>
```

## Output

Let's carry out the following steps to see how the above code works:

- Save the above html code as **jqm\_controlgroup\_selects.html** file in your server root folder.
- Open this HTML file as [http://localhost/jqm\\_controlgroup\\_selects.html](http://localhost/jqm_controlgroup_selects.html) and the following output will be displayed.

### Vertical Select

### Minimized Vertical Select

### Horizontal Select

### Minimized Horizontal Select

## Controlgroup Textinputs

### Description

You can specify the text inputs that allow the users to input data using ***controlgroup-textinput*** class.

### Example

Following example describes the use of controlgroup textinputs in the jQuery Mobile Framework.

```
<!DOCTYPE html>
<head>
<title>Controlgroup Textinputs</title>
<meta name="viewport" content="width=device-width, initial-scale=1">
<link rel="stylesheet"
href="https://code.jquery.com/mobile/1.4.5/jquery.mobile-1.4.5.min.css">
<script src="https://code.jquery.com/jquery-1.11.3.min.js"></script>
<script src="https://code.jquery.com/mobile/1.4.5/jquery.mobile-
1.4.5.min.js"></script>
</head>
<body>
<h2>Controlgroup Textinputs</h2>
<label for="search-control-group">Search Here</label>
<div data-role="controlgroup" data-type="horizontal">
<input type="text" id="search-control-group" data-wrapper-
class="controlgroup-textinput ui-btn"><br><br>
<button>Send</button>
<button>Reset</button>
</div>
</body>
</html>
```

### Output

Let's carry out the following steps to see how the above code works:

- Save the above html code as **jqm\_controlgroup\_textinputs.html** file in your server root folder.
- Open this HTML file as [http://localhost/jqm\\_controlgroup\\_textinputs.html](http://localhost/jqm_controlgroup_textinputs.html) and the following output will be displayed.

## Controlgroup Textinputs

Search Here

## Controlgroup Pre-Rendered Markup

### Description

You can use *pre-rendered* attribute for saving the startup time in the controlgroup.

### Example

Following example describes the use of controlgroup pre-rendered markup in the jQuery Mobile Framework.

```
<!DOCTYPE html>
<head>
<title>Controlgroup Pre-rendered Markup</title>
<meta name="viewport" content="width=device-width, initial-scale=1">
<link rel="stylesheet" href="https://code.jquery.com/mobile/1.4.5/jquery.mobile-1.4.5.min.css">
<link rel="stylesheet" href="style.css">
<script src="https://code.jquery.com/jquery-1.11.3.min.js"></script>
<script src="https://code.jquery.com/mobile/1.4.5/jquery.mobile-1.4.5.min.js"></script>
</head>
<body>
<h2>Pre-rendered Controlgroup</h2>
<div data-role="controlgroup" data-enhanced="true" class="ui-controlgroup-controls ui-controlgroup-vertical ui-corner-all">
  <label for="pre-rendered1">Checkbox 1</label>
  <input type="checkbox" id="pre-rendered1" value="val1">
  <label for="pre-rendered2">Checkbox 2</label>
  <input type="checkbox" id="pre-rendered2" value="val2">
</div>
</body>
</html>
```

## Output

Let's carry out the following steps to see how the above code works:

- **Save the above html code as jqm\_controlgroup\_pre-rendered\_markup.html** file in your server root folder.
- Open this HTML file as [http://localhost/jqm\\_controlgroup\\_pre-rendered\\_markup.html](http://localhost/jqm_controlgroup_pre-rendered_markup.html) and the following output will be displayed.

## Pre-rendered Controlgroup

Checkbox 1  Checkbox 2

## Filterable Widget

---

### Description

By using the **data-filter="true"** attribute, you can filter the children of any element. The child will contain the text, which will be used for filtering by default.

However, you have an alternate option of setting the data-filtertext attribute to a string on any of the child, which will be considered for filtering.

Following table lists filterable types used in jQuery mobile.

| Sr. No. | Filterable Types & Description   |
|---------|--|
| 1       | <b>Basic filter</b><br>You can set <i>data-filter</i> attribute to <i>true</i> on a listview to create a filter for its items of the list.                 |
| 2       | <b>Table filter</b><br>To generate a table filter, set <i>data-filter</i> attribute to <i>true</i> on the element of the table to produce filter for rows. |

|    |   |
|----|---|
| 3  | <b><u>Controlgroup Filter</u></b><br>The controlgroup buttons can also be filtered using <i>data-filter</i> attribute on the elements, which creates the controlgroup.  |
| 4  | <b><u>Filter Collapsible Set</u></b><br>The filter widget can also be used on <i>collapsible set</i> . You have to use <i>data-filter</i> attribute and set to <i>true</i> on the element that creates the collapsible set. |
| 5  | <b><u>Filter Collapsible Set and collapsible children</u></b><br>The filter widget can also be used on collapsible set and collapsible children.  |
| 6  | <b><u>Filter Anything</u></b><br>The filter widget can also be used to filter any element containing another element.   |
| 7  | <b><u>Filter Styling</u></b><br>You can set a specific theme for text field and also provide a placeholder for it.  |
| 8  | <b><u>Filter Reveal</u></b><br>This feature will allow you to build a simple autocomplete easily with local data.   |
| 9  | <b><u>Filter Custom Callback</u></b><br>As with the extension of the listview, the callback function can be provided to the filter or even override the filter entirely on the <i>filterablebeforefilter</i> event.         |
| 10 | <b><u>Pre-rendering</u></b><br>By specifying the <i>data-enhanced="true"</i> attribute, you can prevent the filterable widget executing the filter one time during the startup.   |

## Filterable Basics

---

### Description

You can set **data-filter** attribute to true on a listview to create a filter for its items of the list.

To enter the search string, you have to provide a text field in your markup and filterable makes the use of it by giving a selector to access the text field's value of *data-input* attribute.

### Example

Following example demonstrates the use of basic filterable in jQuery Mobile.

```
<!DOCTYPE html>
<head>
```

```
<title>Filterable basics</title>
<meta name="viewport" content="width=device-width, initial-scale=1">
<link rel="stylesheet"
href="https://code.jquery.com/mobile/1.4.5/jquery.mobile-1.4.5.min.css">
<script src="https://code.jquery.com/jquery-1.11.3.min.js"></script>
<script src="https://code.jquery.com/mobile/1.4.5/jquery.mobile-
1.4.5.min.js"></script>
</head>
<body>
<form class="ui-filterable">
<input id="filterBasic-input" data-type="search">
</form>
<ul data-role="listview" data-filter="true" data-input="#filterBasic-input">
<li>Audi</li>
<li>Benz</li>
<li>BMW</li>
<li>Datsun</li>
<li>Ferrari</li>
<li>Ford</li>
<li>Mahindra</li>
<li>Maruti Suzuki</li>
<li>Renault</li>
<li>Volkswagen</li>
</ul>
</body>
</html>
```

## Output

Let's carry out the following steps to see how the above code works:

- Save the above html code as **filterable\_basic.html** file in your server root folder.
- Open this HTML file as [http://localhost/filterable\\_basic.html](http://localhost/filterable_basic.html) and the following output will be displayed.

- Audi
- Benz
- BMW
- Datsun
- Ferrari
- Ford
- Mahindra
- Maruti Suzuki
- Renault
- Volkswagen

## Table Filter

---

### Description

To generate a table filter, set ***data-filter*** attribute to *true* on the element of the table to produce a filter for rows.

### Example

Following example demonstrates the use of table filter in jQuery Mobile.

```
<!DOCTYPE html>
<head>
    <title>Filterable Table</title>
    <meta name="viewport" content="width=device-width, initial-scale=1">
    <link rel="stylesheet"
    href="https://code.jquery.com/mobile/1.4.5/jquery.mobile-1.4.5.min.css">
    <script src="https://code.jquery.com/jquery-1.11.3.min.js"></script>
    <script src="https://code.jquery.com/mobile/1.4.5/jquery.mobile-
    1.4.5.min.js"></script>
</head>
<body>
    <form>
        <input id="filterTable-input" data-type="search">
    </form>
    <table data-role="table" id="movie-table" data-filter="true" data-
    input="#filterTable-input" class="ui-responsive">
        <thead>
            <tr>
```

```
<th data-priority="1">Rank</th>
<th data-priority="persist">Country</th>
<th data-priority="2">Area sq.km.</th>
<th data-priority="3">Capital</th>
<th data-priority="4">Currency</th>
</tr>
</thead>
<tbody>
<tr>
<th>1</th>
<td>Russia</td>
<td>17,075,200</td>
<td>Moscow</td>
<td>Russian ruble</td>
</tr>
<tr>
<th>2</th>
<td>Canada</td>
<td>9,984,670</td>
<td>Ottawa</td>
<td>Canadian dollar</td>
</tr>
<tr>
<th>3</th>
<td>United States of America</td>
<td>9,826,630</td>
<td>Washington, D.C.</td>
<td>US Dollar</td>
</tr>
<tr>
<th>4</th>
<td>China</td>
<td>9,596,960</td>
<td>Beijing</td>
<td>Renminbi</td>
</tr>
<tr>
```

```

<th>5</th>
<td>Brazil</td>
<td>8,511,965</td>
<td>Brasilia</td>
<td>Brazilian real</td>
</tr>
<tr>
<th>6</th>
<td>Australia</td>
<td>7,686,850</td>
<td>Canberra</td>
<td>Australian dollar</td>
</tr>
<tr>
<th>7</th>
<td>India</td>
<td>3,287,590</td>
<td>New Delhi</td>
<td>Indian Rupee</td>
</tr>
</tbody>
</table>
</body>
</html>

```

## Output

Let's carry out the following steps to see how the above code works:

- Save the above html code as **filterable\_table.html** file in your server root folder.
- Open this HTML file as [http://localhost/filterable\\_table.html](http://localhost/filterable_table.html) and the following output will be displayed.

| Rank | Country                  | Area sq.km. | Capital          | Currency          |
|------|--------------------------|-------------|------------------|-------------------|
| 1    | Russia                   | 17,075,200  | Moscow           | Russian ruble     |
| 2    | Canada                   | 9,984,670   | Ottawa           | Canadian dollar   |
| 3    | United States of America | 9,826,630   | Washington, D.C. | US Dollar         |
| 4    | China                    | 9,596,960   | Beijing          | Renminbi          |
| 5    | Brazil                   | 8,511,965   | Brasilia         | Brazilian real    |
| 6    | Australia                | 7,686,850   | Canberra         | Australian dollar |
| 7    | India                    | 3,287,590   | New Delhi        | Indian Rupee      |

## Controlgroup Filter

### Description

The controlgroup buttons can also be filtered using **data-filter** attribute on elements, which creates the controlgroup. You can also filter the elements based on the text string declared in **data-filtertext** attribute.

### Example

Following example demonstrates the use of filterable controlgroup in jQuery Mobile.

```
<!DOCTYPE html>
<head>
    <title>Filterable controlgroup</title>
    <meta name="viewport" content="width=device-width, initial-scale=1">
    <link rel="stylesheet"
    href="https://code.jquery.com/mobile/1.4.5/jquery.mobile-1.4.5.min.css">
    <script src="https://code.jquery.com/jquery-1.11.3.min.js"></script>
    <script src="https://code.jquery.com/mobile/1.4.5/jquery.mobile-
1.4.5.min.js"></script>
</head>
<body>
    <form>
        <input data-type="search" id="filterControlgroup-input">
    </form>
    <div data-role="controlgroup" data-filter="true" data-
input="#filterControlgroup-input">
        <a href="#" class="ui-btn ui-shadow ui-corner-all">Button 1</a>
        <a href="#" class="ui-btn ui-shadow ui-corner-all" data-
filtertext="two">Button 2</a>
    </div>
</body>

```

```

        <a href="#" class="ui-btn ui-shadow ui-corner-all" data-
filtertext="three">Button 3</a>
        <a href="#" class="ui-btn ui-shadow ui-corner-all">Button 4</a>
        <a href="#" class="ui-btn ui-shadow ui-corner-all">Button 5</a>
    </div>
</body>
</html>

```

## Output

Let's carry out the following steps to see how the above code works:

- Save the above html code as **filterable\_controlgroup.html** file in your server root folder.
- Open this HTML file as [http://localhost/filterable\\_controlgroup.html](http://localhost/filterable_controlgroup.html) and the following output will be displayed.



[Button 1](#) [Button 2](#) [Button 3](#) [Button 4](#) [Button 5](#)

## Filter Collapsible Set

---

### Description

The filter widget can also be used on *collapsible set*. You have to use **data-filter** attribute and set to *true* on the element that creates the collapsible set.

### Example

Following example demonstrates the use of filterable collapsible set in jQuery Mobile.

```

<!DOCTYPE html>
<head>
    <title>Filterable collapsible set</title>
    <meta name="viewport" content="width=device-width, initial-scale=1">

```

```
<link rel="stylesheet"
href="https://code.jquery.com/mobile/1.4.5/jquery.mobile-1.4.5.min.css">
<script src="https://code.jquery.com/jquery-1.11.3.min.js"></script>
<script src="https://code.jquery.com/mobile/1.4.5/jquery.mobile-
1.4.5.min.js"></script>
</head>
<body>
<form>
    <input data-type="search" id="searchForCollapsibleSet">
</form>
<div data-role="collapsibleset" data-filter="true" data-inset="true"
id="collapsiblesetForFilter" data-input="#searchForCollapsibleSet">
    <div data-role="collapsible" data-filtertext="India">
        <h3>India</h3>
        <ul data-role="listview" data-inset="false">
            <li>Bengaluru</li>
            <li>Chennai</li>
            <li>Mumbai</li>
            <li>New Delhi</li>
        </ul>
    </div>
    <div data-role="collapsible" data-filtertext="Australia">
        <h3>Australia</h3>
        <ul data-role="listview" data-inset="false">
            <li>Canberra</li>
            <li>Sydney</li>
            <li>Perth</li>
            <li>Melbourne</li>
        </ul>
    </div>
    <div data-role="collapsible" data-filtertext="United Kingdom">
        <h3>United Kingdom</h3>
        <ul data-role="listview" data-inset="false">
            <li>London</li>
            <li>Birmingham</li>
            <li>Cardiff</li>
            <li>Glasgow</li>
        </ul>
    </div>
</div>
```

```

    </div>
</div>
</body>
</html>
```

## Output

Let's carry out the following steps to see how the above code works:

- Save the above html code as **filterable\_collapsible\_set.html** file in your server root folder.
- Open this HTML file as [http://localhost/filterable\\_collapsible\\_set.html](http://localhost/filterable_collapsible_set.html) and the following output will be displayed.



### India

- Bengaluru
- Chennai
- Mumbai
- New Delhi

### Australia

- Canberra
- Sydney
- Perth
- Melbourne

### United Kingdom

- London
- Birmingham
- Cardiff
- Glasgow

## Filter Collapsible Set & Collapsible Children

### Description

The filter widget can also be used on collapsible set and collapsible children.

## Example

Following example demonstrates the use of filterable collapsible set *and* collapsible children in jQuery Mobile.

```
<!DOCTYPE html>
<head>
    <title>Filterable collapsible set and collapsible children</title>
    <meta name="viewport" content="width=device-width, initial-scale=1">
    <link rel="stylesheet"
    href="https://code.jquery.com/mobile/1.4.5/jquery.mobile-1.4.5.min.css">
    <script src="https://code.jquery.com/jquery-1.11.3.min.js"></script>
    <script src="https://code.jquery.com/mobile/1.4.5/jquery.mobile-
1.4.5.min.js"></script>
</head>
<body>
    <form>
        <input data-type="search" id="searchForCollapsibleSetChildren">
    </form>
    <div data-role="collapsibleset" data-filter="true" data-children="> div, >
div div ul li" data-inset="true" id="collapsiblesetForFilterChildren" data-
input="#searchForCollapsibleSetChildren">
        <div data-role="collapsible" data-filtertext="India Bengaluru Chennai
Mumbai New Delhi">
            <h3>India</h3>
            <ul data-role="listview" data-inset="false">
                <li data-filtertext="India Bengaluru">Bengaluru</li>
                <li data-filtertext="India Chennai">Chennai</li>
                <li data-filtertext="India Mumbai">Mumbai</li>
                <li data-filtertext="India New Delhi">New Delhi</li>
            </ul>
        </div>
        <div data-role="collapsible" data-filtertext="Australia Canberra Sydney
Perth Melbourne">
            <h3>Australia</h3>
            <ul data-role="listview" data-inset="false">
                <li data-filtertext="Australia Canberra">Canberra</li>
                <li data-filtertext="Australia Sydney">Sydney</li>
                <li data-filtertext="Australia Perth">Perth</li>
                <li data-filtertext="Australia Melbourne">Melbourne</li>
            </ul>
        </div>
    </div>

```

```
</ul>
</div>

<div data-role="collapsible" data-filtertext="United Kingdom London
Birmingham Cardiff Glasgow">
    <h3>United Kingdom</h3>
    <ul data-role="listview" data-inset="false">
        <li data-filtertext="United Kingdom Canberra">London</li>
        <li data-filtertext="United Kingdom Birmingham">Birmingham</li>
        <li data-filtertext="United Kingdom Cardiff">Cardiff</li>
        <li data-filtertext="United Kingdom Glasgow">Glasgow</li>
    </ul>
</div>
</div>
</body>
</html>
```

## Output

Let's carry out the following steps to see how the above code works:

- Save the above html code as **filterable\_collapsible\_set\_children.html** file in your server root folder.
- Open this HTML file as [http://localhost/filterable\\_collapsible\\_set\\_children.html](http://localhost/filterable_collapsible_set_children.html) and the following output will be displayed.

## India

- Bengaluru
- Chennai
- Mumbai
- New Delhi

## Australia

- Canberra
- Sydney
- Perth
- Melbourne

## United Kingdom

- London
- Birmingham
- Cardiff
- Glasgow

## Filter Anything

### Description

The filter widget can also be used to filter any element containing another element like a `<div>` element containing `<p>`.

### Example

Following example demonstrates the use of filter anything in jQuery Mobile.

```
<!DOCTYPE html>
<head>
    <title>Filterable Anything</title>
    <meta name="viewport" content="width=device-width, initial-scale=1">
    <link rel="stylesheet"
    href="https://code.jquery.com/mobile/1.4.5/jquery.mobile-1.4.5.min.css">
    <script src="https://code.jquery.com/jquery-1.11.3.min.js"></script>
    <script src="https://code.jquery.com/mobile/1.4.5/jquery.mobile-
    1.4.5.min.js"></script>
```

```

</head>
<body>
  <form>
    <input data-type="search" id="divOfPs-input">
  </form>
  <div class="elements" data-filter="true" data-input="#divOfPs-input">
    <p><strong>Paragraph one</strong> Lorem ipsum dolor sit amet, consectetur
adipiscing elit</p>
    <p><strong>Paragraph two</strong> sed do eiusmod tempor incididunt ut
labore et dolore magna aliqua</p>
    <p><strong>Paragraph three</strong> quis nostrud exercitation ullamco
laboris nisi ut aliquip ex ea commodo consequat</p>
    <p><strong>Paragraph four</strong> Duis aute irure dolor in
reprehenderit in voluptate velit esse cillum dolore eu fugiat nulla
pariatur</p>
    <p><strong>Paragraph five</strong> Excepteur sint occaecat cupidatat non
proident, sunt in culpa qui officia deserunt mollit anim id est laborum</p>
  </div>
</body>
</html>

```

## Output

Let's carry out the following steps to see how the above code works:

- Save the above html code as **filterableAnything.html** file in your server root folder.
- Open this HTML file as <http://localhost/filterableAnything.html> and the following output will be displayed.

**Paragraph one** Lorem ipsum dolor sit amet, consectetur adipiscing elit

**Paragraph two** sed do eiusmod tempor incididunt ut labore et dolore magna aliqua

**Paragraph three** quis nostrud exercitation ullamco laboris nisi ut aliquip ex ea commodo consequat

**Paragraph four** Duis aute irure dolor in reprehenderit in voluptate velit esse cillum dolore eu fugiat nulla pariatur

**Paragraph five** Excepteur sint occaecat cupidatat non proident, sunt in culpa qui officia deserunt mollit anim id est laborum

## Filter Styling

---

### Description

You can set a specific theme for the text field using **data-filter-theme** and also provide a placeholder by using **data-filter-placeholder**.

Additionally, the widget will provide the synchronize options, which will be shared between the widget of the textinput and the widget whose children it filters. This ensures that, the textinput widget option's value is the same as the widget option's value.

For instance, whenever data-inset="true" is set on listview, then the respective textinput widget will also be inset.

### Example

Following example demonstrates the use of *filter styling* in jQuery Mobile.

```
<!DOCTYPE html>
<head>
    <title>Filter Styling</title>
    <meta name="viewport" content="width=device-width, initial-scale=1">
    <link rel="stylesheet"
    href="https://code.jquery.com/mobile/1.4.5/jquery.mobile-1.4.5.min.css">
    <script src="https://code.jquery.com/jquery-1.11.3.min.js"></script>
    <script src="https://code.jquery.com/mobile/1.4.5/jquery.mobile-
    1.4.5.min.js"></script>
</head>
<body>
    <ul data-role="listview" data-filter="true" data-filter-placeholder="Find
    cars..." data-filter-theme="a" data-inset="true">
```

```

<li>Audi</li>
<li>BMW</li>
<li>Ford</li>
<li>Hyundai</li>
<li>Mahindra</li>
<li>porsche</li>
<li>Renault</li>
<li>Toyota</li>
</ul>
</body>
</html>

```

## Output

Let's carry out the following steps to see how the above code works:

- Save the above html code as **filterable\_styling.html** file in your server root folder.
- Open this HTML file as [http://localhost/filterable\\_styling.html](http://localhost/filterable_styling.html) and the following output will be displayed.

- Audi
- BMW
- Ford
- Hyundai
- Mahindra
- porsche
- Renault
- Toyota

## Filter Reveal

---

### Description

Filter reveal feature will allow you to build a simple autocomplete easily with local data.

You can use **data-filter-reveal="true"** attribute to auto hide all of the list items when the text field used for searching is blank.

### Example

Following example demonstrates the use of *filter reveal* in jQuery Mobile.

```

<!DOCTYPE html>
<head>
    <title>Filter Reveal</title>
    <meta name="viewport" content="width=device-width, initial-scale=1">
    <link rel="stylesheet"
    href="https://code.jquery.com/mobile/1.4.5/jquery.mobile-1.4.5.min.css">
    <script src="https://code.jquery.com/jquery-1.11.3.min.js"></script>
    <script src="https://code.jquery.com/mobile/1.4.5/jquery.mobile-
1.4.5.min.js"></script>
</head>
<body>
    <ul data-role="listview" data-filter="true" data-filter-reveal="true" data-
filter-placeholder="Search your products..." data-inset="true">
        <li><a href="#">Electronics</a></li>
        <li><a href="#">Clothing</a></li>
        <li><a href="#">Home and furnitures</a></li>
        <li><a href="#">Media and books</a></li>
        <li><a href="#">Babies and kids</a></li>
        <li><a href="#">Mobile and tablets</a></li>
    </ul>
</body>
</html>

```

## Output

Let's carry out the following steps to see how the above code works:

- Save the above html code as **filterable\_reveal.html** file in your server root folder.
- Open this HTML file as [http://localhost/filterable\\_reveal.html](http://localhost/filterable_reveal.html) and the following output will be displayed.

- [Electronics](#)
- [Clothing](#)
- [Home and furnitures](#)
- [Media and books](#)
- [Babies and kids](#)
- [Mobile and tablets](#)

## Filter Custom Callback

---

### Description

As with the extension of the listview, callback function can be provided to the filter or even override the filter entirely on the *filterablebeforefilter* event. Before the filter fires actually, it takes a delay of 250ms. This avoids the filtering function to run many times while the user is typing.

To create a new default for all filterable widgets, set a custom filtering function, override the *filtercallback* option in widget prototype in a *mobileinit* signal handler as shown in the following code.

```
$( document ).one( "mobileinit", function() {
    $.mobile.filterable.prototype.options.filterCallback = function( index,
    searchValue ) {

        // The this keyword in this function refers to the element
        // for the code which decide whether or not to filter.
        // The true return value shows that the element referred
        // to by this keyword is to be filtered.
        // If returns false specifies that the item is to be displayed.
        //

        // Custom filtering logic goes here.

    });
});
```

By setting the *filterCallback* options, you can set a custom filtering function for a single filterable widget as shown in the following code.

```
$.mobile.document.one( "filterablecreate", "#myFilterable", function() {
    $( "#myFilterable" ).filterable( "option", "filterCallback", function( index,
    searchValue ) {

        // The above example explains the callback function's signature.
        //

        // Custom filtering logic goes here.

    });
});
```

To override the filter completely (for instance, whenever loading the data server-side or from localStorage), bind to the *filterablebeforfilter* event as shown in the following code.

```
$( ".selector" ).on( "filterablebeforfilter", function( e, data ) {
    var value;
```

```

e.preventDefault();
value = data.input.value;
// trigger own request to database
});
});

```

## Filterable Pre-rendering

### Description

The filterable widget executes the filter one time during the startup to ensure that child list returns the entered value in search input. To prevent this, specify the data-enhanced="true" attribute.

Whenever this attribute is set to true, it is assumed that you have correctly applied the ui-screen class to those children which must be hidden initially. The filterable widget uses the search input to determine whether the search input has pre-rendered itself or not.

Following example pre-renders both the search input and the filterable.

### Example

Following example demonstrates the use of filterable *pre*-rendering in jQuery Mobile.

```

<!DOCTYPE html>
<head>
    <title>Filterable Pre-rendering</title>
    <meta name="viewport" content="width=device-width, initial-scale=1">
    <link rel="stylesheet"
    href="https://code.jquery.com/mobile/1.4.5/jquery.mobile-1.4.5.min.css">
    <script src="https://code.jquery.com/jquery-1.11.3.min.js"></script>
    <script src="https://code.jquery.com/mobile/1.4.5/jquery.mobile-
    1.4.5.min.js"></script>
</head>
<body>
    <form>
        <div class="ui-input-search ui-body-inherit ui-corner-all ui-shadow-inset
        ui-input-has-clear">
            <input data-type="search" data-enhanced="true" data-inset="false"
            id="pre-rendered-example-input" placeholder="search cars..." value="or">
        </div>
        <div data-role="controlgroup" data-enhanced="true" data-filter="true"
        data-filter-reveal="true" data-input="#pre-rendered-example-input" class="ui-
        controlgroup ui-controlgroup-vertical ui-corner-all">

```

```

<div class="ui-controlgroup-controls">
    <a href="#" class="ui-btn ui-corner-all ui-shadow ui-shadow ui-
screen-hidden">Audi</a>
    <a href="#" class="ui-btn ui-corner-all ui-shadow ui-first-child
ui-shadow ui-last-child">Ford</a>
    <a href="#" class="ui-btn ui-corner-all ui-shadow ui-shadow ui-
screen-hidden">Hyundai</a>
    <a href="#" class="ui-btn ui-corner-all ui-shadow ui-shadow ui-
screen-hidden">Tata</a>
</div>
</div>
</form>
</body>
</html>

```

## Output

Let's carry out the following steps to see how the above code works:

- Save the above html code as **filterable\_prerendering.html** file in your server root folder.
- Open this HTML file as [http://localhost/filterable\\_prerendering.html](http://localhost/filterable_prerendering.html) and the following output will be displayed.

or  
[Audi](#) [Ford](#) [Hyundai](#) [Tata](#)

## Flip Switch Widget

---

### Description

Flip Switch allows you to turn off/on or true/false the switch by clicking it for boolean style input.

## Example

Following example demonstrates the use of flipswitch widget in the jQuery Mobile.

```
<!DOCTYPE html>
<html>
<head>
    <meta name="viewport" content="width=device-width, initial-scale=1">
    <link rel="stylesheet"
    href="https://code.jquery.com/mobile/1.4.5/jquery.mobile-1.4.5.min.css">
    <script src="https://code.jquery.com/jquery-1.11.3.min.js"></script>
    <script src="https://code.jquery.com/mobile/1.4.5/jquery.mobile-
1.4.5.min.js"></script>

</head>
<body>
<form>
<h2>Flip Switch Examples</h2>

<h3>Basic Flipswitch</h3>
<input type="checkbox" data-role="flipswitch">

<h3>Basic checkbox switch with custom labels</h3>
<input type="checkbox" data-role="flipswitch" data-on-text="YES" data-off-
text="NO" data-wrapper-class="custom-label-flipswitch">

<h3>Basic checkbox switch with custom labels and custom size</h3>
<input type="checkbox" data-role="flipswitch" data-on-text="Yes" data-off-
text="No" data-wrapper-class="custom-size-flipswitch">

<h3>Basic checkbox switch checked</h3>
<input type="checkbox" data-role="flipswitch" checked="">

<h3>Basic select switch</h3>
<select data-role="flipswitch">
    <option>Off</option>
    <option>On</option>
</select>
```

```
<h3>Basic select switch second option selected</h3>
<select id="flip-select-second" name="flip-select" data-role="flipswitch">
    <option>Off</option>
    <option selected="">On</option>
</select>

<h3>Theme</h3>
<input type="checkbox" data-role="flipswitch" data-theme="b">

<h3>Mini</h3>
<input type="checkbox" data-role="flipswitch" data-mini="true">

<h3>No corners</h3>
<input type="checkbox" data-role="flipswitch" data-corners="false">

<h3>Disabled</h3>
<input type="checkbox" data-role="flipswitch" disabled="disabled">

<h3>Label hidden</h3>
<label class="ui-hidden-accessible">Flip toggle switch:</label>
<input type="checkbox" data-role="flipswitch">

<h3>Fieldcontain</h3>
<div class="ui-field-contain">
    <label>Flip Contain:</label>
    <select data-role="flipswitch">
        <option value="off">Off</option>
        <option value="on">On</option>
    </select>
</div>

<h3>Fieldcontain, mini</h3>
<div class="ui-field-contain">
    <label>Flip toggle switch:</label>
    <select data-role="flipswitch" data-mini="true">
        <option value="off">Off</option>
        <option value="on">On</option>
    </select>
</div>
```

```
</select>
</div>
</form>
</body>
</html>
```

## Output

Let's carry out the following steps to see how the above code works:

- Save the above html code as **flipswitch\_basic.html** file in your server root folder.
- Open this HTML file as [http://localhost/flipswitch\\_basic.html](http://localhost/flipswitch_basic.html) and the following output will be displayed.

## Flip Switch Examples

### Basic Flipswitch



### Basic checkbox switch with custom labels



### Basic checkbox switch with custom labels and custom size



### Basic checkbox switch checked



### Basic select switch



### Basic select switch second option selected



### Theme



**Mini****No corners****Disabled****Label hidden**Flip toggle switch: **Fieldcontain**Flip Contain: **Fieldcontain, mini**Flip toggle switch: 

## Listview Widget

---

### Description

The purpose of listview component is to render complex and customized content in lists.

Following table demonstrates the types of listview widget in detail.

| Sr. No. | Types & Description   |
|---------|---|
| 1       | <b><u>Listview</u></b><br>The Listview is used to display a list of items.  |
| 2       | <b><u>Listview Autocomplete</u></b><br>The <i>Filterable widget</i> is used to create autocomplete widget backed by local or remote data. |
| 3       | <b><u>Listview Remote Autocomplete</u></b><br><i>filterablebeforefilter</i> event is used for remote autocomplete data source.            |

|   |   |
|---|---|
| 4 | <b><u>Listview Autodivider Linkbar</u></b><br>Linkbar extension is included for the autodivider linkbar listview.   |
| 5 | <b><u>Listview Autodivider Selector</u></b><br>The <i>autodividersSelector</i> option helps to return a different string.   |
| 6 | <b><u>Nested Listviews</u></b><br>Include the nested listview extension to allow the nested response in jQuery mobile.  |
| 7 | <b><u>Listview Collapsible Item Flat</u></b><br>Collapsibles are the basic and easy widget to display short content.  |
| 8 | <b><u>Listview Collapsible Item Indented</u></b><br>In the tree structure of indented lists, the outermost class is identified by adding class <i>ui-listview-outer</i> .       |
| 9 | <b><u>Listview Responsive Grid</u></b><br>The style is adjusted by using the class <i>ui-li-has-thumb</i> , it makes the image size as same as of the tiles in the grid layout. |

## Listview

---

### Description

The Listview is used to display a list of items. ***data-role="listview"*** attribute is included in the container to display the list in the vertical scrollable list.

Following table demonstrates the types of listview in detail.

| Sr. No. | Types & Description   |
|---------|---|
| 1       | <b><u>Read-only, unordered</u></b><br>Include <i>data-role="listview"</i> attribute to create a simple unordered list.    |
| 2       | <b><u>Read-only, ordered</u></b><br>Create an ordered lists(ol) by including <i>data-role="listview"</i> attribute in it. |
| 3       | <b><u>Linked</u></b><br>When a hyperlink is defined in the list then it is styled as a button.                            |
| 4       | <b><u>Inset</u></b><br>Apply an inset look by including the attribute <i>data-inset="true"</i> .                          |

|    |  |
|----|--|
| 5  | <b><u>Filter</u></b><br>The list can be filtered by adding the <i>data-filter="true"</i> attribute.  |
| 6  | <b><u>Filter reveal</u></b><br>Add the <i>data-filter-reveal="true"</i> attribute to the hide the list item.   |
| 7  | <b><u>List dividers</u></b><br>Add <i>data-role="list-divider"</i> atrribute to any list item to divide into a group of list items.                  |
| 8  | <b><u>Autodividers</u></b><br><i>data-autodividers="true"</i> attribute is included to generate the dividers automatically to any listview.          |
| 9  | <b><u>Count bubbles</u></b><br><i>ui-li-count</i> class define in element will help to include a count indicator in the list item at the right side. |
| 10 | <b><u>Icons: Standard</u></b><br>Add attribute <i>data-icon</i> to set an icon for the particular list item  |
| 11 | <b><u>Icons: 16x16</u></b><br>Include class of <i>ui-li-icon</i> in the image element to add image of 16 X 16 pixels in the list item.               |
| 12 | <b><u>Thumbnails</u></b><br>In list item, the thumbnails is included on the left side.   |
| 13 | <b><u>Split buttons</u></b><br>Include a second link inside the <i>li</i> to create a split list item.   |
| 14 | <b><u>Formatted content</u></b><br>Use heading and paragraph tags to add content in the proper hierarchy format.                                     |
| 15 | <b><u>Theme</u></b><br><i>data-theme</i> attribute is added to include theme for the listview or to the particular list item.                        |
| 16 | <b><u>Forms</u></b><br>In listview, the form element can be inserted for a group presentation.   |
| 17 | <b><u>Collapsible listview</u></b><br><i>data-role="collapsible"</i> attribute is wrapped in the container to make the listview collapsible.         |

|    |  |
|----|--|
| 18 | <b><u>Grouped collapsible with listviews</u></b><br>A group of listview can created inside the collapsible set using attribute <i>data-role="collapsibleset"</i> . |
| 19 | <b><u>Full width collapsible listview</u></b><br><i>data-inset="false"</i> attribute is added to make a full width collapsible set.                                |

## Listview Unordered Widget

---

### Description

Include ***data-role="listview"*** attribute to create a simple unordered list.

### Example

Following example demonstrates the use of unordered list in the jQuery Mobile.

```
<!DOCTYPE html>
<html>
<head>
    <meta name="viewport" content="width=device-width, initial-scale=1">
    <link rel="stylesheet"
    href="https://code.jquery.com/mobile/1.4.5/jquery.mobile-1.4.5.min.css">
    <script src="https://code.jquery.com/jquery-1.11.3.min.js"></script>
    <script src="https://code.jquery.com/mobile/1.4.5/jquery.mobile-
1.4.5.min.js"></script>
</head>
<body>
<h2>Listview Unordered Example</h2>
<ul data-role="listview">
    <li>Chennai</li>
    <li>Bangalore</li>
    <li>Pune</li>
    <li>Mumbai</li>
    <li>Delhi</li>
</ul>
</body>
</html>
```

## Output

Let's carry out the following steps to see how the above code works:

- Save the above html code as **listview\_unordered.html** file in your server root folder.
- Open this HTML file as [http://localhost/listview\\_unordered.html](http://localhost/listview_unordered.html) and the following output will be displayed.

## Listview Unordered Example

- Chennai
- Bangalore
- Pune
- Mumbai
- Delhi

## Listview Ordered

---

### Description

Create an ordered lists(ol) by including ***data-role="listview"*** attribute in it.

### Example

Following example demonstrates the use of ordered list in the jQuery Mobile.

```
<!DOCTYPE html>
<html>
<head>
    <meta name="viewport" content="width=device-width, initial-scale=1">
    <link rel="stylesheet"
    href="https://code.jquery.com/mobile/1.4.5/jquery.mobile-1.4.5.min.css">
    <script src="https://code.jquery.com/jquery-1.11.3.min.js"></script>
    <script src="https://code.jquery.com/mobile/1.4.5/jquery.mobile-
    1.4.5.min.js"></script>
</head>
<body>
<h2>Listview Ordered Example</h2>
<ol data-role="listview">
    <li>Chennai</li>
```

```

<li>Bangalore</li>
<li>Pune</li>
<li>Mumbai</li>
<li>Delhi</li>
</ol>
</body>
</html>

```

## Output

Let's carry out the following steps to see how the above code works:

- Save the above html code as **listview\_ordered.html** file in your server root folder.
- Open this HTML file as [http://localhost/listview\\_ordered.html](http://localhost/listview_ordered.html) and the following output will be displayed.

## Listview Ordered Example

1. Chennai
2. Bangalore
3. Pune
4. Mumbai
5. Delhi

## Listview Linked

---

### Description

When a hyperlink is defined in the list, then it is styled as a button.

### Example

Following example demonstrates the use of linked list in the jQuery Mobile.

```

<!DOCTYPE html>
<html>
<head>
    <meta name="viewport" content="width=device-width, initial-scale=1">
    <link rel="stylesheet"
    href="https://code.jquery.com/mobile/1.4.5/jquery.mobile-1.4.5.min.css">

```

```

<script src="https://code.jquery.com/jquery-1.11.3.min.js"></script>
<script src="https://code.jquery.com/mobile/1.4.5/jquery.mobile-
1.4.5.min.js"></script>
</head>
<body>
<h2>Listview Linked Example</h2>
<ul data-role="listview">
<li><a href="#">Chennai</a></li>
<li><a href="#">Bangalore</a></li>
<li><a href="#">Pune</a></li>
<li><a href="#">Mumbai</a></li>
<li><a href="#">Delhi</a></li>
</ul>
</body>
</html>

```

## Output

Let's carry out the following steps to see how the above code works:

- Save the above html code as **listview\_linked.html** file in your server root folder.
- Open this HTML file as [http://localhost/listview\\_linked.html](http://localhost/listview_linked.html) and the following output will be displayed.

## Listview Linked Example

- [Chennai](#)
- [Bangalore](#)
- [Pune](#)
- [Mumbai](#)
- [Delhi](#)

## Listview Inset

---

### Description

Apply an inset look by including the attribute ***data-inset="true"***. It helps in mixing a listview with other content on a page.

## Example

Following example demonstrates the use of listview inset in the jQuery Mobile.

```
<!DOCTYPE html>
<html>
<head>
    <meta name="viewport" content="width=device-width, initial-scale=1">
    <link rel="stylesheet"
    href="https://code.jquery.com/mobile/1.4.5/jquery.mobile-1.4.5.min.css">
    <script src="https://code.jquery.com/jquery-1.11.3.min.js"></script>
    <script src="https://code.jquery.com/mobile/1.4.5/jquery.mobile-
1.4.5.min.js"></script>
</head>
<body>
<h2>Listview Inset Example</h2>
<ul data-role="listview" data-inset="true">
    <li><a href="">Chennai</a></li>
    <li><a href="">Bangalore</a></li>
    <li><a href="">Pune</a></li>
    <li><a href="">Mumbai</a></li>
    <li><a href="">Delhi</a></li>
</ul>
</body>
</html>
```

## Output

Let's carry out the following steps to see how the above code works:

- Save the above html code as ***listview\_inset.html*** file in your server root folder.
- Open this HTML file as [http://localhost/listview\\_inset.html](http://localhost/listview_inset.html) and the following output will be displayed.

## Listview Inset Example

- [Chennai](#)
- [Bangalore](#)
- [Pune](#)
- [Mumbai](#)
- [Delhi](#)

## Listview Filter

---

### Description

The list can be filtered by adding the **`data-filter="true"`** attribute. A search box is included above the list and behavior to filter out the list items is added. By default, the theme is inherited from the parents for search box.

### Example

Following example demonstrates the use of filter list in the jQuery Mobile.

```
<!DOCTYPE html>
<html>
<head>
    <meta name="viewport" content="width=device-width, initial-scale=1">
    <link rel="stylesheet"
    href="https://code.jquery.com/mobile/1.4.5/jquery.mobile-1.4.5.min.css">
    <script src="https://code.jquery.com/jquery-1.11.3.min.js"></script>
    <script src="https://code.jquery.com/mobile/1.4.5/jquery.mobile-
    1.4.5.min.js"></script>
</head>
<body>
<h2>Listview Filter Example</h2>
<ul data-role="listview" data-inset="true" data-filter="true" data-filter-
placeholder="Search">
    <li><a href="#">Chennai</a></li>
    <li><a href="#">Bangalore</a></li>
    <li><a href="#">Pune</a></li>
    <li><a href="#">Mumbai</a></li>
    <li><a href="#">Delhi</a></li>

```

```

<li><a href="#">Dehradhun</a></li>
<li><a href="#">Punjab</a></li>
</ul>
</body>
</html>

```

## Output

Let's carry out the following steps to see how the above code works:

- Save the above html code as **listview\_filter.html** file in your server root folder.
- Open this HTML file as [http://localhost/listview\\_filter.html](http://localhost/listview_filter.html) and the following output will be displayed.

## Listview Filter Example

- [Chennai](#)
- [Bangalore](#)
- [Pune](#)
- [Mumbai](#)
- [Delhi](#)
- [Dehradhun](#)
- [Punjab](#)

## Listview Filter Reveal

### Description

When the search box is empty, then it will auto hide the complete list item. Add the **data-filter-reveal="true"** attribute to the hide the list item.

### Example

Following example demonstrates the use of filter list reveal in the jQuery Mobile.

```

<!DOCTYPE html>
<html>
<head>
    <meta name="viewport" content="width=device-width, initial-scale=1">
    <link rel="stylesheet"
    href="https://code.jquery.com/mobile/1.4.5/jquery.mobile-1.4.5.min.css">
    <script src="https://code.jquery.com/jquery-1.11.3.min.js"></script>

```

```

<script src="https://code.jquery.com/mobile/1.4.5/jquery.mobile-
1.4.5.min.js"></script>
</head>
<body>
<h2>Listview Filter Reveal Example</h2>
<ul data-role="listview" data-inset="true" data-filter="true" data-filter-
reveal="Search" data-filter-placeholder="Search">
<li><a href="#">Chennai</a></li>
<li><a href="#">Bangalore</a></li>
<li><a href="#">Pune</a></li>
<li><a href="#">Mumbai</a></li>
<li><a href="#">Delhi</a></li>
<li><a href="#">Dehradun</a></li>
<li><a href="#">Punjab</a></li>
</ul>
</body>
</html>

```

## Output

Let's carry out the following steps to see how the above code works:

- Save the above html code as **listview\_filter\_reveal.html** file in your server root folder.
- Open this HTML file as [http://localhost/listview\\_filter\\_reveal.html](http://localhost/listview_filter_reveal.html) and the following output will be displayed.

## Listview Filter Reveal Example

- [Chennai](#)
- [Bangalore](#)
- [Pune](#)
- [Mumbai](#)
- [Delhi](#)
- [Dehradun](#)
- [Punjab](#)

## Listview Dividers

---

### Description

Add ***data-role="list-divider"*** attribute to any list item to divide into a group of list items. ***data-divider-theme*** attribute can be used to include a theme to the list item.

### Example

Following example demonstrates the use of list divider in the jQuery Mobile.

```
<!DOCTYPE html>
<html>
<head>
    <meta name="viewport" content="width=device-width, initial-scale=1">
    <link rel="stylesheet"
    href="https://code.jquery.com/mobile/1.4.5/jquery.mobile-1.4.5.min.css">
    <script src="https://code.jquery.com/jquery-1.11.3.min.js"></script>
    <script src="https://code.jquery.com/mobile/1.4.5/jquery.mobile-
1.4.5.min.js"></script>
</head>
<body>
<h2>List Dividers Example</h2>
<ul data-role="listview" data-inset="true" data-divider-theme="b">
    <li data-role="list-divider">Karnataka</li>
    <li><a href="#">Bangalore</a></li>
    <li><a href="#">Mangalore</a></li>
    <li><a href="#">Hubli</a></li>
    <li data-role="list-divider">Maharastra</li>
    <li><a href="#">Pune</a></li>
    <li><a href="#">Mumbai</a></li>
</ul>
</body>
</html>
```

### Output

Let's carry out the following steps to see how the above code works:

- Save the above html code as ***listview\_list\_dividers.html*** file in your server root folder.

- Open this HTML file as [http://localhost/listview\\_list\\_dividers.html](http://localhost/listview_list_dividers.html) and the following output will be displayed.

## List Dividers Example

- Karnataka
- [Bangalore](#)
- [Mangalore](#)
- [Hubli](#)
- Maharashtra
- [Pune](#)
- [Mumbai](#)

## Listview Autodividers

---

### Description

**data-autodividers="true"** attribute is included to generate the dividers automatically to any listview.

### Example

Following example demonstrates the use of list autodivider in the jQuery Mobile.

```
<!DOCTYPE html>
<html>
<head>
    <meta name="viewport" content="width=device-width, initial-scale=1">
    <link rel="stylesheet"
    href="https://code.jquery.com/mobile/1.4.5/jquery.mobile-1.4.5.min.css">
    <script src="https://code.jquery.com/jquery-1.11.3.min.js"></script>
    <script src="https://code.jquery.com/mobile/1.4.5/jquery.mobile-
1.4.5.min.js"></script>
</head>
<body>
    <h2>List Autodividers Example</h2>
    <ul data-role="listview" data-autodividers="true" data-filter="true" data-
    inset="true">
        <li><a href="#">Dharwad</a></li>
        <li><a href="#">Davangere</a></li>
        <li><a href="#">Mangalore</a></li>
        <li><a href="#">Mumbai</a></li>
    
```

```

<li><a href="#">Pune</a></li>
<li><a href="#">Punjab</a></li>
</ul>
</body>
</html>

```

## Output

Let's carry out the following steps to see how the above code works:

- Save the above html code as **listview\_autodividers.html** file in your server root folder.
- Open this HTML file as [http://localhost/listview\\_autodividers.html](http://localhost/listview_autodividers.html) and the following output will be displayed.

## List Autodividers Example

- [Dharwad](#)
- [Davangere](#)
- [Mangalore](#)
- [Mumbai](#)
- [Pune](#)
- [Punjab](#)

## Listview Count bubbles

---

### Description

*ui-li-count* class define in an element will help to include a count indicator in the list item on the right side. Add **data-count-theme** attribute to set the theme for count bubbles in the list.

### Example

Following example demonstrates the use of count bubble list in the jQuery Mobile.

```

<!DOCTYPE html>
<html>
<head>
    <meta name="viewport" content="width=device-width, initial-scale=1">
    <link rel="stylesheet"
    href="https://code.jquery.com/mobile/1.4.5/jquery.mobile-1.4.5.min.css">

```

```

<script src="https://code.jquery.com/jquery-1.11.3.min.js"></script>
<script src="https://code.jquery.com/mobile/1.4.5/jquery.mobile-
1.4.5.min.js"></script>
</head>
<body>
<h2>Count Bubble List Example</h2>
<ul data-role="listview" data-count-theme="b" data-inset="true">
<li><a href="#">Inbox<span class="ui-li-count">100</span></a></li>
<li><a href="#">Starred<span class="ui-li-count">5</span></a></li>
<li><a href="#">Important<span class="ui-li-count">2</span></a></li>
<li><a href="#">Chats<span class="ui-li-count">6</span></a></li>
<li><a href="#">Sent Mail<span class="ui-li-count">20</span></a></li>
<li><a href="#">Drafts<span class="ui-li-count">12</span></a></li>
</ul>
</body>
</html>

```

## Output

Let's carry out the following steps to see how the above code works:

- Save the above html code as **listview\_count\_bubbles.html** file in your server root folder.
- Open this HTML file as [http://localhost/listview\\_count\\_bubbles.html](http://localhost/listview_count_bubbles.html) and the following output will be displayed.

## Count Bubble List Example

- [Inbox100](#)
- [Starred5](#)
- [Important2](#)
- [Chats6](#)
- [Sent Mail20](#)
- [Drafts12](#)

## Listview Standard Icons

---

### Description

Add attribute ***data-icon*** to set an icon for a particular list item. You can set different types of icons by including the name of any standard icon.

## Example

Following example demonstrates the use of Icon standard in the list in the jQuery Mobile.

```
<!DOCTYPE html>
<html>
<head>
    <meta name="viewport" content="width=device-width, initial-scale=1">
    <link rel="stylesheet"
    href="https://code.jquery.com/mobile/1.4.5/jquery.mobile-1.4.5.min.css">
    <script src="https://code.jquery.com/jquery-1.11.3.min.js"></script>
    <script src="https://code.jquery.com/mobile/1.4.5/jquery.mobile-
1.4.5.min.js"></script>
</head>
<body>
<h2>Icon Standard List Example</h2>
<ul data-role="listview" data-inset="true">
    <li data-icon="custom"><a href="#">Inbox</a></li>
    <li data-icon="gear"><a href="#">Setting</a></li>
    <li data-icon="info"><a href="#">Important</a></li>
    <li data-icon="delete"><a href="#">Chats</a></li>
</ul>
</body>
</html>
```

## Output

Let's carry out the following steps to see how the above code works:

- Save the above html code as **listview\_icons\_standard.html** file in your server root folder.
- Open this HTML file as [http://localhost/listview\\_icons\\_standard.html](http://localhost/listview_icons_standard.html) and the following output will be displayed.

## Icon Standard List Example

- [Inbox](#)
- [Setting](#)
- [Important](#)
- [Chats](#)

## Listview Icons: 16x16

### Description

Include class of `ui-li-icon` in the image element to add image of 16 X 16 pixels in the list item.

### Example

Following example demonstrates the use of Icon 16 X 16 in the list in the jQuery Mobile.

```
<!DOCTYPE html>
<html>
<head>
    <meta name="viewport" content="width=device-width, initial-scale=1">
    <link rel="stylesheet"
    href="https://code.jquery.com/mobile/1.4.5/jquery.mobile-1.4.5.min.css">
    <script src="https://code.jquery.com/jquery-1.11.3.min.js"></script>
    <script src="https://code.jquery.com/mobile/1.4.5/jquery.mobile-
    1.4.5.min.js"></script>
</head>
<body>
<h2>Icon 16X16 List Example</h2>
<ul data-role="listview" data-inset="true">
    <li><a href="#">Apple</a></li>
    <li><a href="#">Banana</a></li>
    <li><a href="#">Mango</a></li>
    <li><a href="#">Strawberry</a></li>

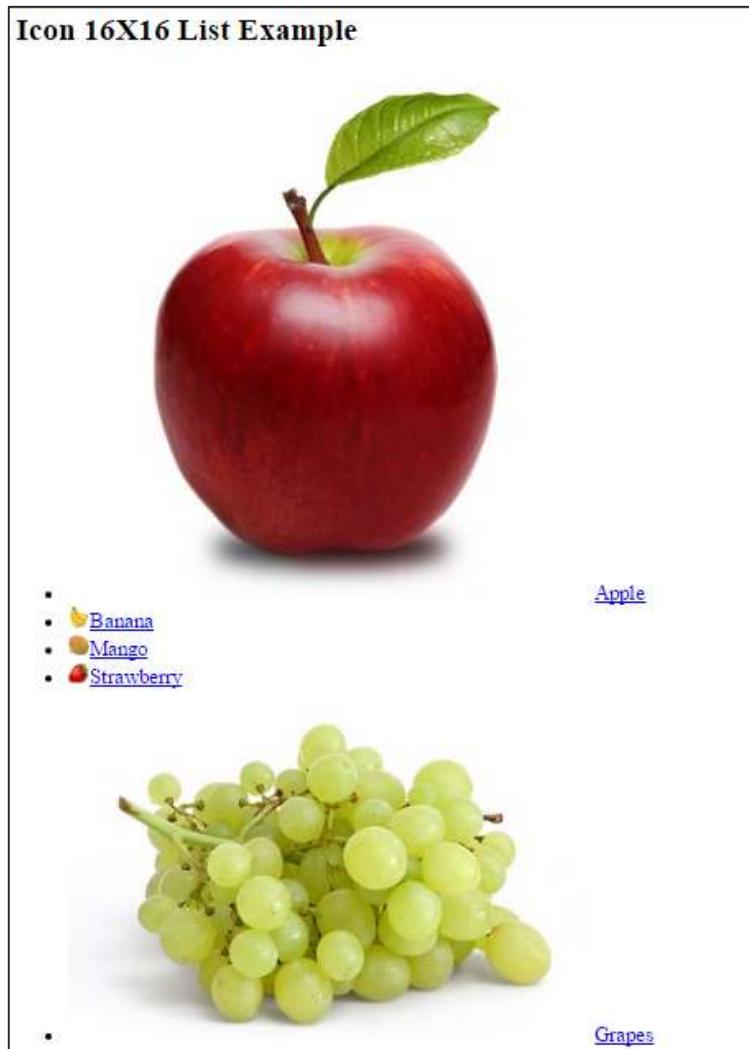
```

```
<li><a href="#">Grapes</a></li>
</ul>
</body>
</html>
```

## Output

Let's carry out the following steps to see how the above code works:

- Save the above html code as **listview\_icons\_16.html** file in your server root folder.
- Open this HTML file as [http://localhost/listview\\_icons\\_16.html](http://localhost/listview_icons_16.html) and the following output will be displayed.



## Listview Thumbnails

---

### Description

In list item, the thumbnails are included on the left side. Insert the image as first child element inside the list item.

### Example

Following example demonstrates the use of thumbnails in the list in the jQuery Mobile.

```
<!DOCTYPE html>
<html>
<head>
    <meta name="viewport" content="width=device-width, initial-scale=1">
    <link rel="stylesheet"
    href="https://code.jquery.com/mobile/1.4.5/jquery.mobile-1.4.5.min.css">
    <script src="https://code.jquery.com/jquery-1.11.3.min.js"></script>
    <script src="https://code.jquery.com/mobile/1.4.5/jquery.mobile-
1.4.5.min.js"></script>
</head>
<body>
<h2>Listview Thumbnail Example</h2>
<ul data-role="listview" data-inset="true">
    <li><a href="#">
        
        <h2>Facebook</h2>
        <p>Mark Zukerburg</p></a>
    </li>
    <li><a href="#">
        
        <h2>Twitter</h2>
        <p>Jack Dorsey</p></a>
    </li>
    <li><a href="#">
        
        <h2>Google+</h2>
        <p>Google</p></a>
    </li>
</ul>
```

```
</body>  
</html>
```

## Output

Let's carry out the following steps to see how the above code works:

- Save the above html code as **listview\_thumbnail.html** file in your server root folder.
- Open this HTML file as [http://localhost/listview\\_thumbnail.html](http://localhost/listview_thumbnail.html) and the following output will be displayed.

## Listview Thumbnail Example



[Facebook](#)

[Mark Zukerburg](#)



[Twitter](#)

[Jack Dorsey](#)



[Google+](#)

[Google](#)

## Listview Split Buttons

### Description

Include a second link inside the ***li*** to create a split list item. ***data-split-icon*** attribute is used to adjust the split button icon and ***data-split-theme*** attribute is used to include theme for the split button.

### Example

Following example demonstrates the use of split button in the list in the jQuery Mobile.

```
<!DOCTYPE html>
<html>
<head>
    <meta name="viewport" content="width=device-width, initial-scale=1">
    <link rel="stylesheet"
    href="https://code.jquery.com/mobile/1.4.5/jquery.mobile-1.4.5.min.css">
    <script src="https://code.jquery.com/jquery-1.11.3.min.js"></script>
    <script src="https://code.jquery.com/mobile/1.4.5/jquery.mobile-
1.4.5.min.js"></script>
</head>
<body>
<h2>Listview Split Button Example</h2>
<ul data-role="listview" data-split-icon="gear" data-split-theme="a" data-
inset="true">
    <li><a href="#">
        
        <h2>Facebook</h2>
        <p>Mark Zukerburg</p></a>
        <a href="#purchase" data-rel="popup" data-position-to="window" data-
transition="pop">Login</a>
    </li>
    <li><a href="#">
        <h2>Twitter</h2>
        <p>Jack Dorsey</p></a>
        <a href="#purchase" data-rel="popup" data-position-to="window" data-
transition="pop">Login</a>
    </li>
    <li><a href="#">
        <h2>Google+</h2>
        <p>Google</p></a>
    </li>
</ul>
```

```

        <a href="#purchase" data-rel="popup" data-position-to="window" data-
transition="pop">Login</a>
    </li>
</ul>

<div data-role="popup" id="purchase" data-theme="a" data-overlay-theme="b"
class="ui-content">
    <h3>Social Media Sites</h3>
    <p>Login in your mobile device</p>
    <a href="index.html" data-rel="back" class="ui-shadow ui-btn ui-btn-b ui-
icon-check ui-btn-icon-left ui-btn-inline ui-mini">LOGIN</a>
    <a href="index.html" data-rel="back" class="ui-shadow ui-btn ui-btn-inline
ui-mini">Cancel</a>
</div>
</body>
</html>
```

## Output

Let's carry out the following steps to see how the above code works:

- Save the above html code as **listview\_split\_buttons.html** file in your server root folder.
- Open this HTML file as [http://localhost/listview\\_split\\_buttons.html](http://localhost/listview_split_buttons.html) and the following output will be displayed.

## Listview Split Button Example



[Facebook](#)

[Mark Zukerburg](#)

[Login](#)



[Twitter](#)

[Jack Dorsey](#)

[Login](#)



[Google+](#)

[Google](#)

[Login](#)

## Social Media Sites

Login in your mobile device

[LOGIN](#) [Cancel](#)

## Listview Formatted Content

### Description

Use heading and paragraph tags to add content in the proper hierarchy format. Include class ***ui-li-aside*** to add supplemental information on the right side of the list item.

### Example

Following example demonstrates the use of Thumbnails in the list in the jQuery Mobile.

```
<!DOCTYPE html>
<html>
<head>
    <meta name="viewport" content="width=device-width, initial-scale=1">
    <link rel="stylesheet"
    href="https://code.jquery.com/mobile/1.4.5/jquery.mobile-1.4.5.min.css">
    <script src="https://code.jquery.com/jquery-1.11.3.min.js"></script>
    <script src="https://code.jquery.com/mobile/1.4.5/jquery.mobile-
1.4.5.min.js"></script>
</head>
<body>
<h2>Listview Formatted Content Example</h2>
<ul data-role="listview" data-inset="true">
    <li data-role="list-divider">Wednesday, January 13, 2016 <span class="ui-li-
count">2</span></li>
    <li><a href="#">
        <h2>John</h2>
        <p><strong>Team Meeting</strong></p>
        <p>Lorem Ipsum is simply dummy text of the printing and typesetting
industry.</p></a>
        <p class="ui-li-aside"><strong>12:48</strong>PM</p>
    </li>
    <li><a href="#">
        <h2>Albert</h2>
        <p><strong>Updation of work</strong></p>
        <p>Lorem Ipsum is simply dummy text of the printing and typesetting
industry.</p></a>
        <p class="ui-li-aside"><strong>4:01</strong>PM</p>
    </li>
    <li data-role="list-divider">Thursday, January 14, 2016 <span class="ui-li-
count">2</span></li>

```

```
<li><a href="#"><h2>Joy</h2>
  <p><strong>New Project Information</strong></p>
  <p>Lorem Ipsum is simply dummy text of the printing and typesetting industry.</p></a>
  <p class="ui-li-aside"><strong>09:25</strong>PM</p>
</li>
</ul>
</body>
</html>
```

## Output

Let's carry out the following steps to see how the above code works:

- Save the above html code as **listview\_formatted\_content.html** file in your server root folder.
- Open this HTML file as [http://localhost/listview\\_formatted\\_content.html](http://localhost/listview_formatted_content.html) and the following output will be displayed.

## Listview Formatted Content Example

- Wednesday, January 13, 2016 2

- [John](#)

[Team Meeting](#)

[Lorem Ipsum is simply dummy text of the printing and typesetting industry.](#)

12:48PM

- [Albert](#)

[Updation of work](#)

[Lorem Ipsum is simply dummy text of the printing and typesetting industry.](#)

4:01PM

- Thursday, January 14, 2016 2

- [Joy](#)

[New Project Information](#)

[Lorem Ipsum is simply dummy text of the printing and typesetting industry.](#)

09:25PM

## Listview Theme

---

### Description

**data-theme** attribute is added to include theme for the listview or to the particular list item.

**data-divider-theme** attribute sets the theme for the list divider.

**data-count-theme** attribute is used to set the theme for the count bubbles.

**data-split-theme** attribute sets the theme for the split button.

## Example

Following example demonstrates the use of theme in the list in the jQuery Mobile.

```

<!DOCTYPE html>
<html>
<head>
    <meta name="viewport" content="width=device-width, initial-scale=1">
    <link rel="stylesheet"
    href="https://code.jquery.com/mobile/1.4.5/jquery.mobile-1.4.5.min.css">
    <script src="https://code.jquery.com/jquery-1.11.3.min.js"></script>
    <script src="https://code.jquery.com/mobile/1.4.5/jquery.mobile-
1.4.5.min.js"></script>
</head>
<body>
<h2>Listview Theme Example</h2>
<h3>Count bubble theme</h3>
<ul data-role="listview" data-inset="true" data-theme="b" data-divider-
theme="a" data-count-theme="a">
    <li data-role="list-divider">Mail Box</li>
    <li><a href="#">Inbox <span class="ui-li-count">100</span></a></li>
    <li><a href="#">Chats<span class="ui-li-count">6</span></a></li>
    <li><a href="#">Sent Mail<span class="ui-li-count">20</span></a></li>
</ul><br/>

<h3>Split Button theme</h3>
<ul data-role="listview" data-split-icon="plus" data-theme="a" data-split-
theme="b" data-inset="true">
    <li><a href="">Facebook</a>
        <a href="#purchase" data-rel="popup" data-position-to="window" data-
transition="pop">Login</a>
    </li>
    <li><a href="">Twitter</a>
        <a href="#purchase" data-theme="a" data-rel="popup" data-position-
to="window" data-transition="pop">Login</a>
    </li>
</ul>
</body>
</html>
```

## Output

Let's carry out the following steps to see how the above code works:

- Save the above html code as **listview\_theme.html** file in your server root folder.
- Open this HTML file as [http://localhost/listview\\_theme.html](http://localhost/listview_theme.html) and the following output will be displayed.

## Listview Theme Example

### Count bubble theme

- Mail Box
- [Inbox 100](#)
- [Chats6](#)
- [Sent Mail20](#)

### Split Button theme

- [Facebook Login](#)
- [Twitter Login](#)

## Listview Forms

---

### Description

In listview, the form element can be inserted for a group presentation.

### Example

Following example demonstrates the use of Thumbnails in the list in the jQuery Mobile.

```
<!DOCTYPE html>
<html>
<head>
    <meta name="viewport" content="width=device-width, initial-scale=1">
    <link rel="stylesheet"
    href="https://code.jquery.com/mobile/1.4.5/jquery.mobile-1.4.5.min.css">
    <script src="https://code.jquery.com/jquery-1.11.3.min.js"></script>
    <script src="https://code.jquery.com/mobile/1.4.5/jquery.mobile-
    1.4.5.min.js"></script>
```

```
</head>
<body>
<h2>Listview Form Example</h2>
<form>
<ul data-role="listview" data-inset="true">
    <li class="ui-field-contain">
        <label for="name">Text</label>
        <input type="text" id="name" data-clear-btn="true">
    </li>
    <li class="ui-field-contain">
        <label for="textarea">Textarea</label>
        <textarea cols="40" rows="8" id="textarea"></textarea>
    </li>
    <li class="ui-field-contain">
        <label for="flip">Flip switch</label>
        <select id="flip" data-role="slider">
            <option value="off">Off</option>
            <option value="on">On</option>
        </select>
    </li>
    <li class="ui-field-contain">
        <label for="slider">Slider</label>
        <input type="range" id="slider" value="0" min="0" max="100" data-highlight="true">
    </li>
    <li class="ui-field-contain">
        <label for="select-choice" class="select">Select option</label>
        <select name="select-choice" id="select-choice-1">
            <option>Standard</option>
            <option>Premium</option>
            <option>Deluxe</option>
        </select>
    </li>
    <li class="ui-body ui-body-b">
        <fieldset class="ui-grid-a">
            <div class="ui-block-a"><button type="submit" class="ui-btn ui-corner-all ui-btn-a">Submit</button></div>
        </fieldset>
    </li>
</ul>
</form>

```

```

<div class="ui-block-b"><button type="Cancel" class="ui-btn ui-corner-all ui-btn-a">Cancel</button></div>
</fieldset>
</li>
</ul>
</form>
</body>
</html>
```

## Output

Let's carry out the following steps to see how the above code works:

- Save the above html code as **listview\_forms.html** file in your server root folder.
- Open this HTML file as [http://localhost/listview\\_forms.html](http://localhost/listview_forms.html) and the following output will be displayed.

## Listview Form Example



## Listview Collapsible

---

### Description

**data-role="collapsible"** attribute is wrapped in the container to make the listview collapsible.

## Example

Following example demonstrates the use of collapsible list in the jQuery Mobile.

```
<!DOCTYPE html>
<html>
<head>
    <meta name="viewport" content="width=device-width, initial-scale=1">
    <link rel="stylesheet"
    href="https://code.jquery.com/mobile/1.4.5/jquery.mobile-1.4.5.min.css">
    <script src="https://code.jquery.com/jquery-1.11.3.min.js"></script>
    <script src="https://code.jquery.com/mobile/1.4.5/jquery.mobile-
1.4.5.min.js"></script>
</head>
<body>
<div data-role="collapsible" data-theme="b" data-content-theme="b">
    <h2>Listview Collapsible Example</h2>
    <ul data-role="listview" data-filter="true">
        <li><a href="#">Chennai</a></li>
        <li><a href="#">Belgaum</a></li>
        <li><a href="#">Bangalore</a></li>
        <li><a href="#">Mumbai</a></li>
        <li><a href="#">Pune</a></li>
        <li><a href="#">Delhi</a></li>
        <li><a href="#">Punjab</a></li>
    </ul>
</div>
</body>
</html>
```

## Output

Let's carry out the following steps to see how the above code works:

- Save the above html code as **listview\_collapsible.html** file in your server root folder.
- Open this HTML file as [http://localhost/listview\\_collapsible.html](http://localhost/listview_collapsible.html) and the following output will be displayed.

## Listview Collapsible Example

- [Chennai](#)
- [Belgaum](#)
- [Bangalore](#)
- [Mumbai](#)
- [Pune](#)
- [Delhi](#)
- [Punjab](#)

## Grouped Collapsible With Listviews

---

### Description

A grouped listview can be created inside the collapsible set using attribute **data-role="collapsibleset"**. Only one list item can be opened at a time.

### Example

Following example demonstrates the use of grouped collapsible with listviews in the jQuery Mobile.

```
<!DOCTYPE html>
<html>
<head>
    <meta name="viewport" content="width=device-width, initial-scale=1">
    <link rel="stylesheet"
    href="https://code.jquery.com/mobile/1.4.5/jquery.mobile-1.4.5.min.css">
    <script src="https://code.jquery.com/jquery-1.11.3.min.js"></script>
    <script src="https://code.jquery.com/mobile/1.4.5/jquery.mobile-
    1.4.5.min.js"></script>
</head>
<body>
<div data-role="collapsibleset" data-theme="a" data-content-theme="b">
    <div data-role="collapsible">
        <h2>Thumbnail</h2>
```

```


- !\[\]\(/jquery\_mobile/images/facebook.jpg\)

## Facebook



Mark Zukerburg

Login
- !\[\]\(/jquery\_mobile/images/twitter.jpg\)

## Twitter



Jack Dorsey

Login
- !\[\]\(/jquery\_mobile/images/google.jpg\)

## Google+



Google

Login



### Social Media Sites



Login in your mobile device

Log In
Cancel



## Formatted Content



- Wednesday, January 13, 2016 <span
class="ui-li-count">2</span>
- ## John


```

```

<p><strong>Team Meeting</strong></p>
<p>Lorem Ipsum is simply dummy text of the printing and typesetting industry.</p></a>
<p class="ui-li-aside"><strong>12:48</strong>PM</p>
</li>
<li><a href="#">
<h2>Albert</h2>
<p><strong>Updation of work</strong></p>
<p>Lorem Ipsum is simply dummy text of the printing and typesetting industry.</p></a>
<p class="ui-li-aside"><strong>4:01</strong>PM</p>
</li>
<li data-role="list-divider">Thursday, January 14, 2016 <span
class="ui-li-count">2</span></li>
<li><a href="#">
<h2>Joy</h2>
<p><strong>New Project Information</strong></p>
<p>Lorem Ipsum is simply dummy text of the printing and typesetting industry.</p></a>
<p class="ui-li-aside"><strong>09:25</strong>PM</p>
</li>
</ul>
</div>
<div data-role="collapsible">
<h2> Filter</h2>
<ul data-role="listview" data-inset="true" data-filter="true" data-
filter-placeholder="Search">
<li><a href="#">Chennai</a></li>
<li><a href="#">Belgaum</a></li>
<li><a href="#">Bangalore</a></li>
<li><a href="#">Pune </a></li>
<li><a href="#">Mumbai</a></li>
<li><a href="#">Delhi</a></li>
</ul>
</div>
</div>
</body>
</html>

```

## Output

Let's carry out the following steps to see how the above code works:

- Save the above html code as **listview\_grouped\_collapsible.html** file in your server root folder.
- Open this HTML file as [http://localhost/listview\\_grouped\\_collapsible.html](http://localhost/listview_grouped_collapsible.html) and the following output will be displayed.

**Thumbnail**

-  [Facebook](#)  
[Mark Zukerburg](#)  
[Login](#)
-  [Twitter](#)  
[Jack Dorsey](#)  
[Login](#)
-  [Google+](#)

[Google](#)

[Login](#)

◆◆◆◆

### Social Media Sites

◆◆◆◆ Login in your mobile device ◆◆◆◆ [Log In](#) ◆◆◆◆ [Cancel](#)

### Formatted Content

- Wednesday, January 13, 2016 2
- **[John](#)**

[Team Meeting](#)

[Lorem Ipsum is simply dummy text of the printing and typesetting industry.](#)

12:48PM

- **[Albert](#)**

[Updation of work](#)

[Lorem Ipsum is simply dummy text of the printing and typesetting industry.](#)

4:01PM

- Thursday, January 14, 2016 2
- **[Joy](#)**

[New Project Information](#)

[Lorem Ipsum is simply dummy text of the printing and typesetting industry.](#)

09:25PM

### Filter

- [Chennai](#)
- [Belgaum](#)
- [Bangalore](#)
- [Pune](#)
- [Mumbai](#)
- [Delhi](#)

## Full Width Collapsible Listview

### Description

**data-inset="false"** attribute is added to make full width collapsible set.

### Example

Following example demonstrates the use of full width collapsible listview in the jQuery Mobile.

```
<!DOCTYPE html>
<html>
<head>
    <meta name="viewport" content="width=device-width, initial-scale=1">
    <link rel="stylesheet"
    href="https://code.jquery.com/mobile/1.4.5/jquery.mobile-1.4.5.min.css">
    <script src="https://code.jquery.com/jquery-1.11.3.min.js"></script>
    <script src="https://code.jquery.com/mobile/1.4.5/jquery.mobile-
1.4.5.min.js"></script>
</head>
<body>
<div data-role="collapsibleset" data-theme="a" data-inset="false">
    <div data-role="collapsible">
        <h2>Thumbnail</h2>
        <ul data-role="listview" data-split-icon="gear" data-split-theme="a"
data-inset="true">
            <li><a href="#">
                
                <h2>Facebook</h2>
                <p>Mark Zukerburg</p></a>
                <a href="#purchase" data-rel="popup" data-position-to="window"
data-transition="pop">Login</a>
            </li>
            <li><a href="#">
                <h2>Twitter</h2>
                <p>Jack Dorsey</p></a>
                <a href="#purchase" data-rel="popup" data-position-to="window"
data-transition="pop">Login</a>
            </li>
            <li><a href="#">
```

```

<h2>Google+</h2>
<p>Google</p></a>
<a href="#purchase" data-rel="popup" data-position-to="window"
data-transition="pop">Login</a>
</li>
</ul>
<div data-role="popup" id="purchase" data-theme="a" data-overlay-
theme="b" class="ui-content" >
<h3>Social Media Sites</h3>
Login in your mobile device
<a href="index.html" data-rel="back" class="ui-shadow ui-btn ui-btn-b
ui-icon-check ui-btn-icon-left ui-btn-inline ui-mini">Log In</a>
<a href="index.html" data-rel="back" class="ui-shadow ui-btn ui-btn-
inline ui-mini">Cancel</a>
</div>
</div>
<div data-role="collapsible">
<h2>Formatted Content</h2>
<ul data-role="listview" data-inset="true">
<li data-role="list-divider">Wednesday, January 13, 2016 <span
class="ui-li-count">2</span></li>
<li><a href="#">
<h2>John</h2>
<p><strong>Team Meeting</strong></p>
<p>Lorem Ipsum is simply dummy text of the printing and typesetting
industry.</p></a>
<p class="ui-li-aside"><strong>12:48</strong>PM</p>
</li>
<li><a href="#">
<h2>Albert</h2>
<p><strong>Updation of work</strong></p>
<p>Lorem Ipsum is simply dummy text of the printing and typesetting
industry.</p></a>
<p class="ui-li-aside"><strong>4:01</strong>PM</p>
</li>
<li data-role="list-divider">Thursday, January 14, 2016 <span
class="ui-li-count">2</span></li>
<li><a href="#"><h2>Joy</h2>
<p><strong>New Project Information</strong></p>

```

```

<p>Lorem Ipsum is simply dummy text of the printing and typesetting industry.</p></a>
    <p class="ui-li-aside"><strong>09:25</strong>PM</p>
</li>
</ul>
</div>
<div data-role="collapsible">
    <h2> Filter</h2>
    <ul data-role="listview" data-inset="true" data-filter="true" data-filter-placeholder="Search">
        <li><a href="#">Chennai</a></li>
        <li><a href="#">Belgaum</a></li>
        <li><a href="#">Bangalore</a></li>
        <li><a href="#">Pune </a></li>
        <li><a href="#">Mumbai</a></li>
        <li><a href="#">Delhi</a></li>
    </ul>
</div>
</div>
</body>
</html>

```

## Output

Let's carry out the following steps to see how the above code works:

- Save the above html code as **listview\_width\_collapsible.html** file in your server root folder.
- Open this HTML file as [http://localhost/listview\\_width\\_collapsible.html](http://localhost/listview_width_collapsible.html) and the following output will be displayed.

## Thumbnail



### [Facebook](#)

[Mark Zukerburg](#)

[Login](#)



### [Twitter](#)

[Jack Dorsey](#)

[Login](#)



## [Google+](#)

[Google](#)

[Login](#)



## Social Media Sites

◆◆◆◆ Login in your mobile device ◆◆◆◆ [Log In](#) ◆◆◆◆ [Cancel](#)

## Formatted Content

- Wednesday, January 13, 2016 2

- [John](#)

### [Team Meeting](#)

[Lorem Ipsum is simply dummy text of the printing and typesetting industry.](#)

12:48PM

- [Albert](#)

[Updation of work](#)

[Lorem Ipsum is simply dummy text of the printing and typesetting industry.](#)

4:01PM

- Thursday, January 14, 2016 2

- [Joy](#)

[New Project Information](#)

[Lorem Ipsum is simply dummy text of the printing and typesetting industry.](#)

09:25PM

### Filter

- [Chennai](#)
- [Belgaum](#)
- [Bangalore](#)
- [Pune](#)
- [Mumbai](#)
- [Delhi](#)

## Listview Autocomplete

---

### Description

The **Filterable widget** is used to create an autocomplete widget backed by local or remote data.

Following table demonstrates the types of listview autocomplete in detail.

| Sr. No. | Types & Description  |
|---------|--|
| 1       | <p><b>Remote data</b><br/> <i>filterablebeforefilter</i> event is used for remote autocomplete data source.</p>  |
| 2       | <p><b>Local data</b><br/> Add the <i>data-filter-reveal="true"</i> attribute to the hide the list item. This helps to build a simple autocomplete with local data.</p> |
| 3       | <p><b>Providing richer search content</b><br/> Add <i>data-filtertext</i> attribute in the particular list item to find against different content.</p>                 |

## Listview Remote Autocomplete

---

### Description

**filterablebeforefilter** event is used for remote autocomplete data source. When the user types in the search query then dynamically a list gets displayed.

### Example

Following example demonstrates the use of remote autocomplete listview in the jQuery Mobile.

```
<!DOCTYPE html>
<html>
<head>
    <meta name="viewport" content="width=device-width, initial-scale=1">
    <link rel="stylesheet"
    href="https://code.jquery.com/mobile/1.4.5/jquery.mobile-1.4.5.min.css">
    <script src="https://code.jquery.com/jquery-1.10.2.min.js"></script>
    <script src="https://code.jquery.com/jquery-1.10.2.js"></script>

    <script src="https://code.jquery.com/mobile/1.4.5/jquery.mobile-
1.4.5.min.js"></script>
    <script>
        $( document ).on( "pagecreate", "#remote", function() {
            $("#flip").on("change", function(){
                if (this.checked) {
                    var inp = $( "#autocomplete" ).filterable( "option", "input" );
                    $(inp).val("Belgaum").trigger("change");
                }
            });

            $( "#autocomplete" ).on( "filterablebeforefilter", function ( e, data ) {
                var $ul = $( this ),
                    $input = $( data.input ),
                    value = $input.val(),
                    html = "";
                $ul.html( "" );
                if ( value && value.length > 2 ) {
                    $ul.html( "<li><div class='ui-loader'><span class='ui-icon ui-
icon-loading'></span></div></li>" );
                }
            });
        });
    </script>
```

```
$ul.listview( "refresh" );
$.ajax({
    url: "http://gd.geobytes.com/AutoCompleteCity",
    dataType: "jsonp",
    crossDomain: true,
    data: {
        q: $input.val()
    }
})
.then( function ( response ) {
    $.each( response, function ( i, val ) {
        html += "<li>" + val + "</li>";
    });
    $ul.html( html );
    $ul.listview( "refresh" );
    $ul.trigger( "updatelayout");
});
});

});

});

</script>
</head>
<body>
<div data-role="page" id="remote">
    <h2>Example of remote autocomplete</h2>

    <form>
        <label for="flip"></label>
        <input type="checkbox" data-role="flipswitch" id="flip" />
    </form>

    <form class="ui-filterable">
        <input id="filter-input" data-type="search" />
    </form>

```

```

<ul id="autocomplete" data-role="listview" data-inset="true" data-
filter="true" data-input="#filter-input"></ul>

<div>
</body>
</html>

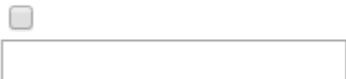
```

## Output

Let's carry out the following steps to see how the above code works:

- Save the above html code as **listview\_remote\_autocomplete.html** file in your server root folder.
- Open this HTML file as [http://localhost/listview\\_remote\\_autocomplete.html](http://localhost/listview_remote_autocomplete.html) and the following output will be displayed.

## Example of remote autocomplete



## Listview Local Data

---

### Description

When the search box is empty, then it will auto hide the complete list item. Add the **data-filter-reveal="true"** attribute to the hide the list item, this helps to build a simple autocomplete with local data.

### Example

Following example demonstrates the use of local data list in the jQuery Mobile.

```

<!DOCTYPE html>
<html>
<head>
  <meta name="viewport" content="width=device-width, initial-scale=1">
  <link rel="stylesheet"
  href="https://code.jquery.com/mobile/1.4.5/jquery.mobile-1.4.5.min.css">
  <script src="https://code.jquery.com/jquery-1.11.3.min.js"></script>

```

```
<script src="https://code.jquery.com/mobile/1.4.5/jquery.mobile-1.4.5.min.js"></script>
</head>
<body>
<h2>Listview local data example</h2>
<form class="ui-filterable">
    <input id="autocomplete-input" data-type="search" placeholder="Search Cities">
</form>
<ul data-role="listview" data-filter="true" data-filter-reveal="true" data-input="#autocomplete-input">
    <li><a href="#">Chennai</a></li>
    <li><a href="#">Belgaum</a></li>
    <li><a href="#">Bangalore</a></li>
    <li><a href="#">Mumbai</a></li>
    <li><a href="#">Pune</a></li>
    <li><a href="#">Delhi</a></li>
    <li><a href="#">Punjab</a></li>
    <li><a href="#">Jhansi</a></li>
    <li><a href="#">Kolkata</a></li>
    <li><a href="#">Kerala</a></li>
    <li><a href="#">Mangalore</a></li>
</ul>

<h2>Inset Listview local example</h2>
<form class="ui-filterable">
    <input id="inset-input" data-type="search" placeholder="Search Cities">
</form>
<ul data-role="listview" data-inset="true" data-filter="true" data-filter-reveal="true" data-input="#inset-input">
    <li><a href="#">Chennai</a></li>
    <li><a href="#">Belgaum</a></li>
    <li><a href="#">Bangalore</a></li>
    <li><a href="#">Mumbai</a></li>
    <li><a href="#">Pune</a></li>
    <li><a href="#">Delhi</a></li>
    <li><a href="#">Punjab</a></li>
    <li><a href="#">Jhansi</a></li>
```

```

<li><a href="#">Kolkata</a></li>
<li><a href="#">Kerala</a></li>
<li><a href="#">Mangalore</a></li>
</ul>
</body>
</html>

```

## Output

Let's carry out the following steps to see how the above code works:

- Save the above html code as **listview\_local\_data.html** file in your server root folder.
- Open this HTML file as [http://localhost/listview\\_local\\_data.html](http://localhost/listview_local_data.html) and the following will be displayed.

### Listview local data example

- [Chennai](#)
- [Belgaum](#)
- [Bangalore](#)
- [Mumbai](#)
- [Pune](#)
- [Delhi](#)
- [Punjab](#)
- [Jhansi](#)
- [Kolkata](#)
- [Kerala](#)
- [Mangalore](#)

### Inset Listview local example

- [Chennai](#)
- [Belgaum](#)
- [Bangalore](#)
- [Mumbai](#)
- [Pune](#)
- [Delhi](#)
- [Punjab](#)
- [Jhansi](#)
- [Kolkata](#)
- [Kerala](#)
- [Mangalore](#)

## Listview Providing Richer Search Content

### Description

Add ***data-filtertext*** attribute in a particular list item, to find against different content which is popular by one or many names. It helps to match with that content when the word is inserted in the search box.

### Example

Following example demonstrates the use of providing richer search content in list in the jQuery Mobile.

```
<!DOCTYPE html>
<html>
<head>
    <meta name="viewport" content="width=device-width, initial-scale=1">
    <link rel="stylesheet"
    href="https://code.jquery.com/mobile/1.4.5/jquery.mobile-1.4.5.min.css">
    <script src="https://code.jquery.com/jquery-1.11.3.min.js"></script>
    <script src="https://code.jquery.com/mobile/1.4.5/jquery.mobile-
1.4.5.min.js"></script>
</head>
<body>
<h2>Listview providing richer search content example</h2>
<form class="ui-filterable">
    <input id="autocomplete-input" data-type="search" placeholder="Search
    Cities">
</form>
<ul data-role="listview" data-filter="true" data-filter-reveal="true" data-
input="#autocomplete-input">
    <li data-filtertext="IT companies"><a href="#">Wipro</a></li>
    <li data-filtertext="IT companies"><a href="#">Infosys</a></li>
    <li data-filtertext="IT companies"><a href="#">Oracle</a></li>
    <li data-filtertext="Automobiles company"><a href="#">Tata</a></li>
    <li data-filtertext="Automobiles company"><a href="#">Volvo </a></li>
    <li data-filtertext="Financial Company"><a href="#">LIC Finance
    Limited</a></li>
    <li data-filtertext="Financial Company"><a href="#">L & T Finance
    Limited</a></li>
    <li data-filtertext="IT companies"><a href="#">TCS</a></li>
    <li data-filtertext="Financial Company"><a href="#">Birla Global Finance
    Limited</a></li>
```

```

<li data-filtertext="Automobiles company"><a href="#">Mahindra </a></li>
<li data-filtertext="IT companies"><a href="#">Tech Mahinder</a></li>
</ul>
</body>
</html>

```

## Output

Let's carry out the following steps to see how the above code works:

- Save the above html code as **listview\_richer\_search\_content.html** file in your server root folder.
- Open this HTML file as [http://localhost/listview\\_richer\\_search\\_content.html](http://localhost/listview_richer_search_content.html) and the following output will be displayed.
- Type IT, it, comp, companies, auto, etc. to see the results.

## Listview providing richer search content example

Search Cities

- [Wipro](#)
- [Infosys](#)
- [Oracle](#)
- [Tata](#)
- [Volvo](#)
- [LIC Finance Limited](#)
- [L & T Finance Limited](#)
- [TCS](#)
- [Birla Global Finance Limited](#)
- [Mahindra](#)
- [Tech Mahinder](#)

## Listview Remote Autocomplete

### Description

**filterablebeforefilter** event is used for remote autocomplete data source. When the user types in the search query, then dynamically a list gets displayed.

### Example

Following example demonstrates the use of remote autocomplete listview in the jQuery Mobile.

```
<!DOCTYPE html>
<html>
<head>
    <meta name="viewport" content="width=device-width, initial-scale=1">
    <link rel="stylesheet"
    href="https://code.jquery.com/mobile/1.4.5/jquery.mobile-1.4.5.min.css">
    <script src="https://code.jquery.com/jquery-1.10.2.min.js"></script>
    <script src="https://code.jquery.com/jquery-1.10.2.js"></script>

    <script src="https://code.jquery.com/mobile/1.4.5/jquery.mobile-
1.4.5.min.js"></script>
    <script>
        $( document ).on( "pagecreate", "#remote", function() {
            $("#flip").on("change", function(){
                if (this.checked) {
                    var inp = $( "#autocomplete" ).filterable( "option", "input" );
                    $(inp).val("Belgaum").trigger("change");
                }
            });

            $( "#autocomplete" ).on( "filterablebeforefilter", function ( e, data ) {
                var $ul = $( this ),
                    $input = $( data.input ),
                    value = $input.val(),
                    html = "";
                $ul.html( "" );
                if ( value && value.length > 2 ) {
                    $ul.html( "<li><div class='ui-loader'><span class='ui-icon ui-
icon-loading'></span></div></li>" );
                    $ul.listview( "refresh" );
                    $.ajax({
                        url: "http://gd.geobytes.com/AutoCompleteCity",
                        dataType: "jsonp",
                        crossDomain: true,
                        data: {
                            q: $input.val()
                        }
                    });
                }
            });
        });
    </script>

```

```

        }
    })
    .then( function ( response ) {
        $.each( response, function ( i, val ) {
            html += "<li>" + val + "</li>";
        });
        $ul.html( html );
        $ul.listview( "refresh" );
        $ul.trigger( "updatelayout" );
    });
}
});

});
</script>
</head>
<body>
<div data-role="page" id="remote">
    <h2>Example of remote autocomplete</h2>

    <form>
        <label for="flip"></label>
        <input type="checkbox" data-role="flipswitch" id="flip" />
    </form>

    <form class="ui-filterable">
        <input id="filter-input" data-type="search" />
    </form>

        <ul id="autocomplete" data-role="listview" data-inset="true" data-
filter="true" data-input="#filter-input"></ul>

    <div>
</body>
</html>

```

## Output

Let's carry out the following steps to see how the above code works:

- Save the above html code as **listview\_remote\_autocomplete.html** file in your server root folder.
- Open this HTML file as [http://localhost/listview\\_remote\\_autocomplete.html](http://localhost/listview_remote_autocomplete.html) and the following output will be displayed.

## Example of remote autocomplete



## Listview Autodivider Linkbar

### Description

Linkbar extension is included for the autodivider linkbar listview. It adds a bar fixed at the right side of the screen, which helps to navigate to a specific part of listview very easily.

### Example

Following example demonstrates the use of Listview Autodivider Linkbar in the jQuery Mobile.

```
<!DOCTYPE html>
<head>
    <meta name="viewport" content="width=device-width, initial-scale=1">
    <link rel="stylesheet"
    href="https://code.jquery.com/mobile/1.4.5/jquery.mobile-1.4.5.min.css">
    <link rel="stylesheet" href=" http://demos.jquerymobile.com/1.4.5/listview-
autodividers-linkbar/autodividers-linkbar.css">
    <script src="https://code.jquery.com/jquery-1.11.3.min.js"></script>
    <script src="https://code.jquery.com/jquery-1.11.3.js"></script>
    <script src="https://code.jquery.com/mobile/1.4.5/jquery.mobile-
1.4.5.min.js"></script>
    <script src="http://demos.jquerymobile.com/1.4.5/listview-autodividers-
linkbar/autodividers-linkbar.js"></script>
</head>
<body>
```

```
<div data-role="page" id="demo-page">
  <div data-role="header">
    <h2>Listview Autodivider Linkbar</h2>
    <a href="#" class="jqm-search-link ui-shadow ui-btn ui-btn-icon-notext ui-corner-all ui-icon-search ui-nodisc-icon ui-alt-icon ui-btn-right">Search</a>
  </div>
  <div role="main" class="ui-content">
    <div id="sorter">
      <ul data-role="listview">
        <li><span>A</span></li>
        <li><span>B</span></li>
        <li><span>C</span></li>
        <li><span>D</span></li>
        <li><span>E</span></li>
        <li><span>F</span></li>
        <li><span>G</span></li>
        <li><span>H</span></li>
      </ul>
    </div>
    <ul data-role="listview" data-autodividers="true" id="sortedList">
      <li><a href="#">Albania</a></li>
      <li><a href="#">Algeria</a></li>
      <li><a href="#">Angola</a></li>
      <li><a href="#">Bahamas</a></li>
      <li><a href="#">Barbados</a></li>
      <li><a href="#">Belgium</a></li>
      <li><a href="#">Canada</a></li>
      <li><a href="#">Cameroon</a></li>
      <li><a href="#">China</a></li>
      <li><a href="#">Denmark</a></li>
      <li><a href="#">Dominica</a></li>
      <li><a href="#">Djibouti</a></li>
      <li><a href="#">Ecuador</a></li>
      <li><a href="#">Egypt</a></li>
      <li><a href="#">Estonia</a></li>
      <li><a href="#">Fiji</a></li>
    </ul>
  </div>
</div>
```

```
<li><a href="#">Finland</a></li>
<li><a href="#">France</a></li>
<li><a href="#">Georgia</a></li>
<li><a href="#">Germany</a></li>
<li><a href="#">Greece</a></li>
<li><a href="#">Haiti</a></li>
<li><a href="#">Hong Kong</a></li>
<li><a href="#">Hungary</a></li>
</ul>
</div>
<div data-role="footer">
    <h4>Footer</h4>
</div>
</div>
</body>
</html>
```

## Output

Let's carry out the following steps to see how the above code works:

- Save the above html code as **listview\_autodivider\_linkbar.html** file in your server root folder.
- Open this HTML file as [http://localhost/listview\\_autodivider\\_linkbar.html](http://localhost/listview_autodivider_linkbar.html) and the following output will be displayed.

## Listview Autodivider Linkbar

[Search](#)

- A
- B
- C
- D
- E
- F
- G
- H
  
- [Albania](#)
- [Algeria](#)
- [Angola](#)
- [Bahamas](#)
- [Barbados](#)
- [Belgium](#)
- [Canada](#)
- [Cameroon](#)
- [China](#)
- [Denmark](#)
- [Dominica](#)
- [Djibouti](#)
- [Ecuador](#)
- [Egypt](#)
- [Estonia](#)
- [Fiji](#)
- [Finland](#)
- [France](#)
- [Georgia](#)
- [Germany](#)
- [Greece](#)
- [Haiti](#)
- [Hong Kong](#)
- [Hungary](#)

[Footer](#)

## Listview Autodivider Selector

---

### Description

The first character of the list item is used by the autodivider plugin by default. The **autodividersSelector** option helps to return a different string.

## Example

Following example demonstrates the use of listview autodivider selector in the jQuery Mobile.

```
<!DOCTYPE html>
<html>
<head>
    <meta name="viewport" content="width=device-width, initial-scale=1">
    <link rel="stylesheet"
    href="https://code.jquery.com/mobile/1.4.5/jquery.mobile-1.4.5.min.css">
    <script src="https://code.jquery.com/jquery-1.11.3.min.js"></script>
    <script src="https://code.jquery.com/mobile/1.4.5/jquery.mobile-
1.4.5.min.js"></script>
    <script>
        $(document).ready(function() {
            var dateList = $("#selector");
            for(var i=0, len=selector.length; i<len; i++) {
                dateList.append("<li>" + selector[i] + "</li>");
            }

            dateList.listview({
                autodividersSelector: function ( li ) {
                    var d = new Date(li.text());
                    return (d.getMonth()+1)+ "/" + d.getDate() + "/" +
d.getFullYear();
                }
            }).listview("refresh");
        });
    </script>
</head>
<body>
<ul data-role="listview" data-inset="true" id="selector" data-
autodividers="true">
    <li><a href="#">01/01/2016 09:15:00</a></li>
    <li><a href="#">01/05/2016 10:00:00</a></li>
    <li><a href="#">01/05/2016 12:45:00</a></li>
    <li><a href="#">01/12/2016 12:00:00</a></li>
    <li><a href="#">01/12/2016 18:30:00</a></li>
    <li><a href="#">01/16/2016 14:00:00</a></li>

```

```
</ul>
</body>
</html>
```

## Output

Let's carry out the following steps to see how the above code works:

- Save the above html code as **listview\_autodivider\_selector.html** file in your server root folder.
- Open this HTML file as [http://localhost/listview\\_autodivider\\_selector.html](http://localhost/listview_autodivider_selector.html) and the following output will be displayed.

- [01/01/2016 09:15:00](#)
- [01/05/2016 10:00:00](#)
- [01/05/2016 12:45:00](#)
- [01/12/2016 12:00:00](#)
- [01/12/2016 18:30:00](#)
- [01/16/2016 14:00:00](#)

## Nested Listviews

---

### Description

Include the nested listview extension to allow the nested response in jQuery mobile and set the *childpages* option to false for a specific list. jQuery mobile 1.3 restores the nested listview extension in jQuery mobile 1.4.

### Example

Following example demonstrates the use of listview nested in the jQuery Mobile.

```
<!DOCTYPE html>
<html>
<head>
    <meta name="viewport" content="width=device-width, initial-scale=1">
    <link rel="stylesheet"
    href="https://code.jquery.com/mobile/1.4.5/jquery.mobile-1.4.5.min.css">
    <script src="https://code.jquery.com/jquery-1.11.3.min.js"></script>
    <script src="https://code.jquery.com/mobile/1.4.5/jquery.mobile-
    1.4.5.min.js"></script>
```

```
<script>
(function( $, window, undefined ) {
$.widget( "mobile.listview", $.mobile.listview, {
    options: {
        childPages: true,
        page: "<div data-role='page'></div>",
        header: "<div data-role='header'><a href='#' data-rel='back'>Back</a><h1></h1></div>",
        content: "<div class='ui-content'></div>"
    },
    _create: function(){
        this._super();
        if( this.options.childPages ) {
            this._setupChildren();
        }
    },
    _setupChildren: function() {
        this._attachBindings();
        this.element.find( "ul" )
        .css( "display","none" )
        .parent()
        .addClass("ui-btn ui-btn-icon-right ui-icon-carat-r");
    },
    _attachBindings: function() {
        this._on({
            "click": "_handleSubpageClick"
        });
        this._on( "body", {
            "pagechange": function(){
                if ( this.opening === true ) {
                    this.open = true;
                    this.opening = false;
                } else if ( this.open === true ) {
                    this.newPage.remove();
                    this.open = false;
                }
            }
        });
    }
});
```

```

        }
    });
},
_handleSubpageClick: function( event ) {
    if( $(event.target).closest( "li" ).children( "ul" ).length == 0 ) {
        return;
    }
    this.opening = true;
    this.newPage = $( this.options.page ).uniqueId();
    this.nestedList = $( event.target ).children( "ul" )
        .clone().attr( "data-" + $.mobile.ns + "role", "listview" )
        .css( "display", "block" );
    this.pageName = (
        $( event.target.childNodes[0] ).text().replace(/\s+/g,
        '').length > 0 )?
        $( event.target.childNodes[0] ).text() :
    $( event.target.childNodes[1] ).text();
    this.pageID = this.newPage.attr( "id" );

    // Build new page
    this.newPage.append(
        $( this.options.header ).find( "h1" ).text( this.pageName ).end()
    )
    .append(
        $( this.options.content )
    )

    .find( "div.ui-content" ).append( this.nestedList );
    $( "body" ).append( this.newPage );
    $( "body" ).pagecontainer( "change", "#" + this.pageID );
}
);
})( jQuery, this );
</script>
</head>
<body>
<ul data-role="listview" data-inset="true">

```

```
<li data-role="list-divider">State Names</li>
<li>
    Karnataka
    <ul>
        <li>Bangalore</li>
        <li>Belgaum</li>
        <li>Hubli</li>
        <li>Mangalore</li>
        <li>Dharwad</li>
    </ul>
</li>
<li>
    Maharashtra
    <ul>
        <li>Mumbai</li>
        <li>Pune</li>
        <li>Satara</li>
        <li>Sangali</li>
        <li>Thane</li>
    </ul>
</li>
<li>
    Tamil Nadu
    <ul>
        <li>Chennai</li>
        <li>Coimbator</li>
        <li>Madurai</li>
        <li>Vellore</li>
        <li>Ooty</li>
    </ul>
</li>
</ul>
</body>
</html>
```

## Output

Let's carry out the following steps to see how the above code works:

- Save the above html code as **listview\_nested\_lists.html** file in your server root folder.
- Open this HTML file as [http://localhost/listview\\_nested\\_lists.html](http://localhost/listview_nested_lists.html) and the following output will be displayed.

- State Names
- Karnataka
  - Bangalore
  - Belgaum
  - Hubli
  - Mangalore
  - Dharwad
- Maharashtra
  - Mumbai
  - Pune
  - Satara
  - Sangali
  - Thane
- Tamil Nadu
  - Chennai
  - Coimbator
  - Madurai
  - Vellore
  - Ooty

## Listview Collapsible Item Flat

---

### Description

Collapsibles are the basic and easy widget to display short content. Using custom CSS, you can create collapsible list items to collapse borders and padding.

### Example

Following example demonstrates the use of listview collapsible list items in the jQuery Mobile.

```
<!DOCTYPE html>
<html>
<head>
  <meta name="viewport" content="width=device-width, initial-scale=1">
  <link rel="stylesheet"
  href="https://code.jquery.com/mobile/1.4.5/jquery.mobile-1.4.5.min.css">
```

```
<script src="https://code.jquery.com/jquery-1.11.3.min.js"></script>
<script src="https://code.jquery.com/mobile/1.4.5/jquery.mobile-
1.4.5.min.js"></script>
</head>
<body>
<h2>Regular listview</h2>
<ul data-role="listview">
<li data-role="collapsible" data-iconpos="right">
<h2>Choose Your State</h2>
<fieldset data-role="controlgroup" data-type="horizontal">
<label>Karnataka<input type="checkbox"></label>
<label>Maharashtra<input type="checkbox"></label>
<label>Tamil Nadu<input type="checkbox"></label>
</fieldset>
</li>
<li data-role="collapsible" data-iconpos="right" data-inset="false">
<h2>Tamil Nadu</h2>
<ul data-role="listview" data-theme="b">
<li><a href="#">Chennai</a></li>
<li><a href="#">Coimbator</a></li>
<li><a href="#">Madurai</a></li>
<li><a href="#">Vellore</a></li>
<li><a href="#">Ooty</a></li>
</ul>
</li>

<li data-role="collapsible" data-iconpos="right" data-inset="false">
<h2>Karnataka</h2>
<ul data-role="listview" data-theme="b">
<li><a href="#">Bangalore</a></li>
<li><a href="#">Belgaum</a></li>
<li><a href="#">Hubli</a></li>
<li><a href="#">Mangalore</a></li>
<li><a href="#">Dharwad</a></li>
</ul>
</li>
</li>
```

```
<a href="#"><h2>Maharastra</h2></a>
</li>
</ul>

<h2>Inset listview</h2>
<ul data-role="listview" data-inset="true" data-shadow="false">
    <li data-role="collapsible" data-iconpos="right" data-inset="false">
        <h2>Choose Your State</h2>
        <fieldset data-role="controlgroup" data-type="horizontal">
            <label>Karnataka<input type="checkbox"></label>
            <label>Maharastra<input type="checkbox"></label>
            <label>Tamil Nadu<input type="checkbox"></label>
        </fieldset>
    </li>
    <li data-role="collapsible" data-iconpos="right" data-inset="false">
        <h2>Tamil Nadu</h2>
        <ul data-role="listview">
            <li><a href="#">Chennai</a></li>
            <li><a href="#">Coimbator</a></li>
            <li><a href="#">Madurai</a></li>
            <li><a href="#">Vellore</a></li>
            <li><a href="#">Ooty</a></li>
        </ul>
    </li>
    <li data-role="collapsible" data-iconpos="right" data-inset="false">
        <h2>Karnataka</h2>
        <ul data-role="listview">
            <li><a href="#">Bangalore</a></li>
            <li><a href="#">Belgaum</a></li>
            <li><a href="#">Hubli</a></li>
            <li><a href="#">Mangalore</a></li>
            <li><a href="#">Dharwad</a></li>
        </ul>
    </li>
    <li>
        <a href="#"><h2>Maharastra</h2></a>
    </li>
</ul>
```

```
</body>  
</html>
```

## Output

Let's carry out the following steps to see how the above code works:

- Save the above html code as **listview\_collapsible\_list\_items.html** file in your server root folder.
- Open this HTML file as [http://localhost/listview\\_collapsible\\_list\\_items.html](http://localhost/listview_collapsible_list_items.html) and the following output will be displayed.

### Regular listview

- Choose Your State
  - Karnataka  Maharashtra  Tamil Nadu
- Tamil Nadu
  - [Chennai](#)
  - [Coimbator](#)
  - [Madurai](#)
  - [Vellore](#)
  - [Ooty](#)
- Karnataka
  - [Bangalore](#)
  - [Belgaum](#)
  - [Hubli](#)
  - [Mangalore](#)
  - [Dharwad](#)
- [Maharastra](#)

## Inset listview

- Choose Your State

Karnataka  Maharastra  Tamil Nadu

- Tamil Nadu

- [Chennai](#)
- [Coimbator](#)
- [Madurai](#)
- [Vellore](#)
- [Ooty](#)

- Karnataka

- [Bangalore](#)
- [Belgaum](#)
- [Hubli](#)
- [Mangalore](#)
- [Dharwad](#)

- [Maharashtra](#)

## Listview Collapsible Item Indented

---

### Description

In the tree structure of indented lists, the outermost class is identified by adding class ***ui-listview-outer***.

### Example

Following example demonstrates the use of listview collapsible item indented in the jQuery Mobile.

```
<!DOCTYPE html>
<html>
<head>
    <meta name="viewport" content="width=device-width, initial-scale=1">
    <link rel="stylesheet"
    href="https://code.jquery.com/mobile/1.4.5/jquery.mobile-1.4.5.min.css">
    <script src="https://code.jquery.com/jquery-1.11.3.min.js"></script>
    <script src="https://code.jquery.com/mobile/1.4.5/jquery.mobile-
    1.4.5.min.js"></script>
</head>
<body>
```

```

<h2>Regular Indent listview</h2>
<ul data-role="listview" class="ui-listview-outer">
    <li data-role="collapsible" data-iconpos="right">
        <h2>Choose Your State</h2>
        <fieldset data-role="controlgroup" data-type="horizontal">
            <label>Karnataka<input type="checkbox"></label>
            <label>Maharastra<input type="checkbox"></label>
            <label>Tamil Nadu<input type="checkbox"></label>
        </fieldset>
    </li>
    <li data-role="collapsible" data-iconpos="right">
        <h2>Tamil Nadu</h2>
        <ul data-role="listview" data-theme="b" data-inset="true">
            <li><a href="#">Chennai</a></li>
            <li><a href="#">Coimbator</a></li>
            <li><a href="#">Madurai</a></li>
            <li><a href="#">Vellore</a></li>
            <li><a href="#">Ooty</a></li>
        </ul>
    </li>

    <li data-role="collapsible" data-iconpos="right">
        <h2>Karnataka</h2>
        <ul data-role="listview" data-theme="b" data-inset="true">
            <li><a href="#">Bangalore</a></li>
            <li><a href="#">Belgaum</a></li>
            <li><a href="#">Hubli</a></li>
            <li><a href="#">Mangalore</a></li>
            <li><a href="#">Dharwad</a></li>
        </ul>
    </li>
    <li>
        <a href="#"><h2>Maharastra</h2></a>
    </li>
</ul>
<h2>Inset Indent listview</h2>
<ul data-role="listview" data-inset="true" class="ui-listview-outer">

```

```

<li data-role="collapsible" data-iconpos="right" data-inset="true">
    <h2>Choose Your State</h2>
    <fieldset data-role="controlgroup" data-type="horizontal">
        <label>Karnataka<input type="checkbox"></label>
        <label>Maharashtra<input type="checkbox"></label>
        <label>Tamil Nadu<input type="checkbox"></label>
    </fieldset>
</li>
<li data-role="collapsible" data-iconpos="right">
    <h2>Tamil Nadu</h2>
    <ul data-role="listview" data-inset="true">
        <li><a href="#">Chennai</a></li>
        <li><a href="#">Coimbator</a></li>
        <li><a href="#">Madurai</a></li>
        <li><a href="#">Vellore</a></li>
        <li><a href="#">Ooty</a></li>
    </ul>
</li>
<li data-role="collapsible" data-iconpos="right">
    <h2>Karnataka</h2>
    <ul data-role="listview" data-inset="true">
        <li><a href="#">Bangalore</a></li>
        <li><a href="#">Belgaum</a></li>
        <li><a href="#">Hubli</a></li>
        <li><a href="#">Mangalore</a></li>
        <li><a href="#">Dharwad</a></li>
    </ul>
</li>
<li>
    <a href="#"><h2>Maharashtra</h2></a>
</li>
</ul>
</li>
</body>
</html>

```

## Output

Let's carry out the following steps to see how the above code works:

- Save the above html code **as listview\_collapsible\_itemIndented.html** file in your server root folder.
- Open this HTML file as [http://localhost/listview\\_collapsible\\_itemIndented.html](http://localhost/listview_collapsible_itemIndented.html) and the following output will be displayed.

## Regular Indent listview

- Choose Your State

Karnataka  Maharashtra  Tamil Nadu

- Tamil Nadu

- [Chennai](#)
- [Coimbator](#)
- [Madurai](#)
- [Vellore](#)
- [Ooty](#)

- Karnataka

- [Bangalore](#)
- [Belgaum](#)
- [Hubli](#)
- [Mangalore](#)
- [Dharwad](#)

- [Maharastra](#)

## Inset Indent listview

- Choose Your State

Karnataka  Maharastra  Tamil Nadu

- Tamil Nadu

- [Chennai](#)
- [Coimbator](#)
- [Madurai](#)
- [Vellore](#)
- [Ooty](#)

- Karnataka

- [Bangalore](#)
- [Belgaum](#)
- [Hubli](#)
- [Mangalore](#)
- [Dharwad](#)

- [Maharastra](#)

## Listview Responsive Grid

---

### Description

The style is adjusted using the class ***ui-li-has-thumb***. It makes the image size as same as of the tiles in the grid layout.

### Example

Following example demonstrates the use of responsive grid listview in the jQuery Mobile.

```
<!DOCTYPE html>
<html>
<head>
    <meta name="viewport" content="width=device-width, initial-scale=1">
    <link rel="stylesheet"
    href="https://code.jquery.com/mobile/1.4.5/jquery.mobile-1.4.5.min.css">
```

```
<link rel="stylesheet" href="http://demos.jquerymobile.com/1.4.5/listview-grid/listview-grid.css">
<script src="https://code.jquery.com/jquery-1.11.3.min.js"></script>
<script src="https://code.jquery.com/jquery-1.11.3.js"></script>
<script src="https://code.jquery.com/mobile/1.4.5/jquery.mobile-1.4.5.min.js"></script>
</head>
<body>
<div data-role="page" data-theme="a" class="my-page" id="demo-page">
    <div data-role="header">
        <h2>Responsive Grid Example</h2>
    </div>
    <div role="main" class="ui-content">
        <ul data-role="listview" data-inset="true">
            <li><a href="#">
                
                <h2>Fruits</h2>
                <p>Apple</p>
                <p class="ui-li-aside">Apple</p></a>
            </li>
            <li><a href="#">
                
                <h2>Fruits</h2>
                <p>Cherry</p>
                <p class="ui-li-aside">Cherry</p></a>
            </li>
            <li><a href="#">
                
                <h2>Fruits</h2>
                <p>Grapes</p>
                <p class="ui-li-aside">Grapes</p></a>
            </li>
            <li><a href="#">
                
                <h2>Fruits</h2>
                <p>Watermelon</p>
                <p class="ui-li-aside">Watermelon</p></a>
            </li>
        </ul>
    </div>
</div>
```

```
</li>
</ul>
</div>
</div>
</body>
</html>
```

## Output

Let's carry out the following steps to see how the above code works:

- Save the above html code as **listview\_responsive\_grid.html** file in your server root folder.
- Open this HTML file as [http://localhost/listview\\_responsive\\_grid.html](http://localhost/listview_responsive_grid.html) and the following output will be displayed.

## Responsive Grid Example



### Fruits

[Apple](#)

[Apple](#)



## Fruits

[Cherry](#)

[Cherry](#)



- ## Fruits

- [Grapes](#)

- [Grapes](#)



- ## Fruits

- [Watermelon](#)

- [Watermelon](#)

## Loader Widget

---

### Description

jQuery Mobile provides different ways of loading states to an element.

Following table lists some of the loading states, which are supported by the jQuery Mobile.

| Sr. No. | Type & Description   |
|---------|--|
| 1       | <b><u>Standard Loader</u></b><br>You can specify the loading state with icon only, text only, or both icon and text. |
| 2       | <b><u>Custom HTML</u></b><br>You can add any HTML to the loader.   |
| 3       | <b><u>Theme</u></b><br>Sets different type of theme on the loader.   |

## Standard Loader Widget

---

### Description

You can specify the loading state with icon only, text only, or both icon and text by using ***data-textonly*** and ***data-textvisible*** attributes.

### Example

Following example describes the use of *standard loader* in the jQuery Mobile Framework.

```
<!DOCTYPE html>
<head>
<meta name="viewport" content="width=device-width, initial-scale=1">
<link rel="stylesheet"
href="https://code.jquery.com/mobile/1.4.5/jquery.mobile-1.4.5.min.css">
<script src="https://code.jquery.com/jquery-1.11.3.min.js"></script>
<script src="https://code.jquery.com/mobile/1.4.5/jquery.mobile-
1.4.5.min.js"></script>
<script>
$( document ).on( "click", ".load_msg", function() {
  var $this = $( this ),
    theme = $this.jqmData( "theme" ) || $.mobile.loader.prototype.options.theme,
```

```

msgText = $this.jqmData( "msgtext" ) ||
$.mobile.loader.prototype.options.text,
textVisible = $this.jqmData( "textvisible" ) ||
$.mobile.loader.prototype.options.textVisible,
textonly = (!$this.jqmData( "textonly" ));
$.mobile.loading( "show", {
    text: msgText,
    textVisible: textVisible,
    theme: theme,
    textonly: textonly
});
})
.on( "click", ".hide_page", function() {
    $.mobile.loading( "hide" );
});
</script>
</head>
<body>
<button class="load_msg" data-textonly="false" data-textvisible="false" data-
inline="true">Default Loading</button>
<button class="load_msg" data-textonly="false" data-textvisible="true" data-
inline="true">Loading with Text</button>
<button class="load_msg" data-textonly="true" data-textvisible="true" data-
msgtext="Text only loader" data-inline="true">Display only Text</button>
<button class="hide_page" data-inline="true" data-icon="delete">Hide the
Text</button>
</body>
</html>

```

## Output

Let's carry out the following steps to see how the above code works:

- Save the above html code as **jqm\_standard\_loader.html** file in your server root folder.
- Open this HTML file as [http://localhost/jqm\\_standard\\_loader.html](http://localhost/jqm_standard_loader.html) and the following output will be displayed.

[Default Loading](#) [Loading with Text](#) [Display only Text](#) [Hide the Text](#)

## Custom HTML Widget

---

### Description

You can add any HTML to the loader using the **data-html** attribute.

### Example

Following example describes the use of custom html *widget* in the jQuery Mobile Framework.

```
<!DOCTYPE html>
<head>
<meta name="viewport" content="width=device-width, initial-scale=1">
<link rel="stylesheet"
href="https://code.jquery.com/mobile/1.4.5/jquery.mobile-1.4.5.min.css">
<script src="https://code.jquery.com/jquery-1.11.3.min.js"></script>
<script src="https://code.jquery.com/mobile/1.4.5/jquery.mobile-
1.4.5.min.js"></script>
<script>
$( document ).on( "click", ".load_msg", function() {
    var $this = $( this ),
        theme = $this.jqmData( "theme" ) || $.mobile.loader.prototype.options.theme,
        msgText = $this.jqmData( "msgtext" ) ||
        $.mobile.loader.prototype.options.text,
        textVisible = $this.jqmData( "textvisible" ) ||
        $.mobile.loader.prototype.options.textVisible,
        textonly = (!$this.jqmData( "textonly" ) );
    $.mobile.loading( "show", {
        text: msgText,
        textVisible: textVisible,
```

```

        theme: theme,
        textonly: textonly
    });
})
.on( "click", ".hide_page", function() {
    $.mobile.loading( "hide" );
});
</script>
</head>
<body>
<button class="load_msg" data-theme="b" data-textonly="true" data-
textvisible="true" data-msgtext="This is custom loading..." data-inline="true"
data-html=<span class="ui-bar ui-shadow ui-overlay-d ui-corner-all"><h2>its loading...</h2></span>This is
Custom HTML</button>
<button class="hide_page" data-inline="true" data-icon="delete">It hides the
page</button>
</body>
</html>

```

## Output

Let's carry out the following steps to see how the above code works:

- Save the above html code as **jqm\_custom\_html.html** file in your server root folder.
- Open this HTML file as [http://localhost/jqm\\_custom\\_html.html](http://localhost/jqm_custom_html.html) and the following output will be displayed.



## Theme Widget

---

### Description

You can set different types of theme on the loader using the ***data-theme*** attribute.

### Example

Following example describes the use of theme widget in the jQuery Mobile Framework.

```
<!DOCTYPE html>
<head>
<meta name="viewport" content="width=device-width, initial-scale=1">
<link rel="stylesheet"
href="https://code.jquery.com/mobile/1.4.5/jquery.mobile-1.4.5.min.css">
<script src="https://code.jquery.com/jquery-1.11.3.min.js"></script>
<script src="https://code.jquery.com/mobile/1.4.5/jquery.mobile-
1.4.5.min.js"></script>
<script>
$( document ).on( "click", ".load_msg", function() {
    var $this = $( this ),
        theme = $this.jqmData( "theme" ) || $.mobile.loader.prototype.options.theme,
        msgText = $this.jqmData( "msgtext" ) ||
        $.mobile.loader.prototype.options.text,
        textVisible = $this.jqmData( "textvisible" ) ||
        $.mobile.loader.prototype.options.textVisible,
        textonly = (!$this.jqmData( "textonly" ));
    $.mobile.loading( "show", {
        text: msgText,
        textVisible: textVisible,
        theme: theme,
        textonly: textonly
    });
})
.on( "click", ".hide_page", function() {
    $.mobile.loading( "hide" );
});
</script>
</head>
<body>
```

```

<button class="load_msg" data-theme="a" data-textvisible="true" data-
msgtext="It will load the theme a" data-inline="true">Theme A</button>
<button class="load_msg" data-theme="b" data-textvisible="true" data-
msgtext="It will load the theme b" data-inline="true">Theme B</button>
<button class="hide_page" data-inline="true" data-icon="delete">It hides the
loading of theme</button>
</body>
</html>

```

## Output

Let's carry out the following steps to see how the above code works:

- Save the above html code as **jqm\_theme.html** file in your server root folder.
- Open this HTML file as [http://localhost/jqm\\_theme.html](http://localhost/jqm_theme.html) and the following output will be displayed.

Theme A   Theme B   It hides the loading of theme

## Navbar Widget

---

### Description

The navbar widget is a set of buttons, which links you to other web pages or sections. The jQuery mobile provides a basic navbar widget consisting of buttons upto 5 and an optional icon in a bar.

Following table lists navbar types used in jQuery mobile.

| Sr. No. | Navbar Types & Description  |
|---------|---|
| 1       | <b><u>Navbar basics</u></b><br>Navbar can be created by using the <b><i>data-role="navbar"</i></b> attribute.   |
| 2       | <b><u>Multi-row</u></b><br>Whenever a navbar consists of more than 5 items, all the items will be wrapped to multiple lines two across.                                       |
| 3       | <b><u>Navbars in headers</u></b><br>You can add a navbar to the top of the page by adding the navbar container within the header block.                                       |
| 4       | <b><u>Navbars in footers</u></b><br>To add the navbar to the page's bottom, wrap the navbar within a container with a <b><i>data-role="footer"</i></b> attribute.             |
| 5       | <b><u>Persistent</u></b><br>The persistent navbar works like a tab bar that appears to be fixed as you browse across the pages.   |
| 6       | <b><u>Icons</u></b><br>To add icons to your navbar use <b><i>data-icon</i></b> attribute to each anchor.  |
| 7       | <b><u>Icon position</u></b><br>The navbar icons can be positioned on the navbar container instead of for every single link.   |
| 8       | <b><u>3rd party icon sets</u></b><br>You can add third party icons to your navbar. To achieve this, just add custom styles to link to the icons and place them in the navbar. |
| 9       | <b><u>Themes</u></b><br>Just like buttons, the theme swatch will be inherited by the navbars from their parent container.   |
| 10      | <b><u>Navbars with button elements</u></b><br>Instead of using links inside the navbar, you can also use button elements.   |

## Navbar Basics

---

### Description

Navbar can be created using the **data-role="navbar"** attribute. When a navbar button is clicked, it obtains the active state. The ui-btn-active class will be removed from all navbar anchors and is added to the link which is activated.

If this is a link to other page, then, after the transition completion class will be removed again. Depending upon the navbar items, the width of the browser will be evenly divided among the items.

### Example

Following example demonstrates the use of basic navbar in jQuery Mobile.

```
<!DOCTYPE html>
<html>
<head>
    <meta name="viewport" content="width=device-width, initial-scale=1">
    <link rel="stylesheet"
    href="https://code.jquery.com/mobile/1.4.5/jquery.mobile-1.4.5.min.css">
    <script src="https://code.jquery.com/jquery-1.11.3.min.js"></script>
    <script src="https://code.jquery.com/mobile/1.4.5/jquery.mobile-
1.4.5.min.js"></script>
</head>
<body>
    <p>Navbar will be rendered 100% with one item.</p>
    <div data-role="navbar">
        <ul>
            <li><a href="#" class="ui-btn-active">One</a></li>
        </ul>
    </div>
    <p>Navbar will be rendered 50% with two items.</p>
    <div data-role="navbar">
        <ul>
            <li><a href="#" class="ui-btn-active">One</a></li>
            <li><a href="#">Two</a></li>
        </ul>
    </div>
    <p>Each button will take 1/3 width of the browser window when navbar has 3
    items.</p>
```

```

<div data-role="navbar">
    <ul>
        <li><a href="#" class="ui-btn-active">One</a></li>
        <li><a href="#">Two</a></li>
        <li><a href="#">Three</a></li>
    </ul>
</div>

<p>Each button will take 1/4 width of the browser window when navbar has 4 items.</p>

<div data-role="navbar">
    <ul>
        <li><a href="#" class="ui-btn-active">One</a></li>
        <li><a href="#">Two</a></li>
        <li><a href="#">Three</a></li>
        <li><a href="#">Four</a></li>
    </ul>
</div>

<p>Each button will take 1/5 width of the browser window when navbar has 5 items.</p>

<div data-role="navbar">
    <ul>
        <li><a href="#" class="ui-btn-active">One</a></li>
        <li><a href="#">Two</a></li>
        <li><a href="#">Three</a></li>
        <li><a href="#">Four</a></li>
        <li><a href="#">Five</a></li>
    </ul>
</div>
</body>
</html>

```

## Output

Let's carry out the following steps to see how the above code works:

- Save the above html code as **navbar\_basic.html** file in your server root folder.
- Open this HTML file as [http://localhost/navbar\\_basic.html](http://localhost/navbar_basic.html) and the following output will be displayed.

Navbar will be rendered 100% with one item.

- [One](#)

Navbar will be rendered 50% with two items.

- [One](#)
- [Two](#)

Each button will take 1/3 width of the browser window when navbar has 3 items.

- [One](#)
- [Two](#)
- [Three](#)

Each button will take 1/4 width of the browser window when navbar has 4 items.

- [One](#)
- [Two](#)
- [Three](#)
- [Four](#)

Each button will take 1/5 width of the browser window when navbar has 5 items.

- [One](#)
- [Two](#)
- [Three](#)
- [Four](#)
- [Five](#)

## Navbar Multi-row

---

### Description

Whenever a navbar consists of more than 5 items, all the items will be wrapped to multiple lines two across.

### Example

Following example demonstrates the use of multi-row navbar in jQuery Mobile.

```
<!DOCTYPE html>
<html>
<head>
<title>Multi-row navbar</title>
```

```

<meta name="viewport" content="width=device-width, initial-scale=1">
<link rel="stylesheet"
href="https://code.jquery.com/mobile/1.4.5/jquery.mobile-1.4.5.min.css">
<script src="https://code.jquery.com/jquery-1.11.3.min.js"></script>
<script src="https://code.jquery.com/mobile/1.4.5/jquery.mobile-
1.4.5.min.js"></script>
</head>
<body>
<div data-role="navbar">
<ul>
<li><a href="#" class="ui-btn-active">One</a></li>
<li><a href="#">Two</a></li>
<li><a href="#">Three</a></li>
<li><a href="#">Four</a></li>
<li><a href="#">Five</a></li>
<li><a href="#">Six</a></li>
<li><a href="#">Seven</a></li>
<li><a href="#">Eight</a></li>
</ul>
</div>
</body>
</html>

```

## Output

Let's carry out the following steps to see how the above code works:

- Save the above html code as **navbar\_multi\_row.html** file in your server root folder.
- Open this HTML file as [http://localhost/navbar\\_multi\\_row.html](http://localhost/navbar_multi_row.html) and the following output will be displayed.

- [One](#)
- [Two](#)
- [Three](#)
- [Four](#)
- [Five](#)
- [Six](#)
- [Seven](#)
- [Eight](#)

## Navbars in Headers

### Description

You can add the navbar to the top of the page by adding the navbar container within the header block.

### Example

Following example demonstrates the use of navbars in headers in jQuery Mobile.

```
<!DOCTYPE html>
<html>
<head>
    <title>Navbars in headers</title>
    <meta name="viewport" content="width=device-width, initial-scale=1">
    <link rel="stylesheet"
    href="https://code.jquery.com/mobile/1.4.5/jquery.mobile-1.4.5.min.css">
    <script src="https://code.jquery.com/jquery-1.11.3.min.js"></script>
    <script src="https://code.jquery.com/mobile/1.4.5/jquery.mobile-
1.4.5.min.js"></script>
</head>
<body>
    <div data-role="header" style="overflow:hidden;">
        <h2>This is Header</h2>
        <a href="#" data-icon="gear" class="ui-btn-right">Options</a>
        <div data-role="navbar">
            <ul>
                <li><a href="#" class="ui-btn-active">Home</a></li>
                <li><a href="#">About Us</a></li>
                <li><a href="#">Contact Us</a></li>
            </ul>
        </div>
    </div>
</body>
</html>
```

## Output

Let's carry out the following steps to see how the above code works:

- Save the above html code as **navbar\_headers.html** file in your server root folder.
- Open this HTML file as [http://localhost/navbar\\_headers.html](http://localhost/navbar_headers.html) and the following output will be displayed.



## This is Header

### Options

- [Home](#)
- [About Us](#)
- [Contact Us](#)

---

## Navbars in Footers

### Description

To add the navbar to the page's bottom wrap, use the navbar within a container with a **data-role="footer"** attribute.

### Example

Following example demonstrates the use of navbar in footer in jQuery Mobile.

```
<!DOCTYPE html>
<html>
<head>
    <title>Navbars in footers</title>
    <meta name="viewport" content="width=device-width, initial-scale=1">
    <link rel="stylesheet"
    href="https://code.jquery.com/mobile/1.4.5/jquery.mobile-1.4.5.min.css">
    <script src="https://code.jquery.com/jquery-1.11.3.min.js"></script>
    <script src="https://code.jquery.com/mobile/1.4.5/jquery.mobile-
    1.4.5.min.js"></script>
</head>
<body>
    <div data-role="footer" style="overflow:hidden;">
        <h2>This is Footer</h2>
    </div>
</body>
</html>
```

```

<div data-role="navbar">
    <ul>
        <li><a href="#" class="ui-btn-active">Home</a></li>
        <li><a href="#">About Us</a></li>
        <li><a href="#">Contact Us</a></li>
    </ul>
</div>
</div>
</body>
</html>

```

## Output

Let's carry out the following steps to see how above code works:

- Save the above html code as **navbar\_footers.html** file in your server root folder.
- Open this HTML file as [http://localhost/navbar\\_footers.html](http://localhost/navbar_footers.html) and the following output will be displayed.

## This is Footer

- [Home](#)
- [About Us](#)
- [Contact Us](#)

## Navbars Persistent

---

### Description

The persistent navbar works like a tab bar that appears to be fixed as you browse across the pages. Adding class ui-btn-active to the anchor in your markup, you can set active state for an item upon the navbar initialization.

Whenever navbar is clicked, the content of the page changes and the persistent header and footer toolbar remains fixed. These toolbars must be placed on each page.

### Example

Following example demonstrates the use of navbar persistent in jQuery Mobile.

```
<!DOCTYPE html>
<html>
<head>
    <meta name="viewport" content="width=device-width, initial-scale=1">
    <link rel="stylesheet"
    href="https://code.jquery.com/mobile/1.4.5/jquery.mobile-1.4.5.min.css">
    <script src="https://code.jquery.com/jquery-1.11.3.min.js"></script>
    <script src="https://code.jquery.com/mobile/1.4.5/jquery.mobile-
1.4.5.min.js"></script>
<script>
    $(function() {
        $( "[data-role='navbar']" ).navbar();
        $( "[data-role='header'], [data-role='footer']" ).toolbar();
    });

    // Update the toolbar's contents
    $( document ).on( "pagecontainerchange", function() {
        // Each of the 4 pages in this example has a data-title attribute
        // which value is equal to the nav button's text
        // For e.g., on first page: <div data-role="page" data-
title="Details">

        var current = $( ".ui-page-active" ).jqmData( "title" );
        // Change the heading
        $( "[data-role='header'] h1" ).text( current );
        // Remove active class
        $( "[data-role='navbar'] a.ui-btn-active" ).removeClass( "ui-btn-
active" );
        // Add active class to current nav button
        $( "[data-role='navbar'] a" ).each(function() {
            if ( $( this ).text() === current ) {
                $( this ).addClass( "ui-btn-active" );
            }
        });
    });
</script>
</head>
<body>
    <div data-role="header" data-position="fixed" data-theme="a">
```

```

        <a href="" data-rel="back" class="ui-btn ui-btn-left ui-alt-icon ui-nodisc-icon ui-corner-all ui-btn-icon-notext ui-icon-carat-1">Back</a>

        <h1>Details</h1>
    </div>
    <div data-role="page" data-title="Details" class="jqm-demos">
        <div role="main" class="ui-content jqm-content jqm-fullwidth">
            <h1>jQuery Mobile</h1>

            <p>The jQuery Mobile is a user interface framework which is built on jQuery core and used for developing responsive websites or applications that are accessible on mobile, tablet and desktop devices. It uses features of both jQuery and jQueryUI to provide API features for mobile web applications.</p>
            <ul class="list">
                <li><p>The jQuery Mobile creates web applications in such a way that it will work the same way on the mobile, tablet and desktop devices.</p></li>
                <li><p>The jQuery Mobile is compatible with other frameworks such as <i>PhoneGap</i>, <i>Whitelight</i> etc.</p></li>
                <li><p>The jQuery Mobile provides set of touch friendly form inputs and UI widgets.</p></li>
                <li><p>The progressive enhancement brings you functionality to all mobile, tablet and desktop platforms and adds efficient page loads and wider device support.</p></li>
            </ul>
        </div>
    </div>
    <div data-role="footer" data-position="fixed" data-theme="a">
        <div data-role="navbar">
            <ul>
                <li><a href="/jquery_mobile/src/navbar_persistent.html" data-prefetch="true" data-transition="flip">Details</a></li>
                <li><a href="/jquery_mobile/src/page_one.html" data-prefetch="true" data-transition="fade">Friends</a></li>
                <li><a href="/jquery_mobile/src/page_two.html" data-prefetch="true" data-transition="slide">Messages</a></li>
                <li><a href="/jquery_mobile/src/page_three.html" data-prefetch="true" data-transition="turn">Emails</a></li>
            </ul>
        </div>
    </div>
</body>
</html>

```

## Output

Let's carry out the following steps to see how the above code works:

- Save the above html code as **navbar\_persistent.html** file in your server root folder.
- Open this HTML file as [http://localhost/navbar\\_persistent.html](http://localhost/navbar_persistent.html) and the following output will be displayed.

[Back](#)

## Details

# jQuery Mobile

The jQuery Mobile is a user interface framework which is built on jQuery core and used for developing responsive websites or applications that are accessible on mobile, tablet and desktop devices. It uses features of both jQuery and jQueryUI to provide API features for mobile web applications.

- The jQuery Mobile creates web applications in such a way that it will work the same way on the mobile, tablet and desktop devices.
- The jQuery Mobile is compatible with other frameworks such as *PhoneGap*, *Whitelight* etc.
- The jQuery Mobile provides set of touch friendly form inputs and UI widgets.
- The progressive enhancement brings you functionality to all mobile, tablet and desktop platforms and adds efficient page loads and wider device support.
- [Details](#)
- [Friends](#)
- [Messages](#)
- [Emails](#)

## Navbars Icons

---

### Description

To add icons to your navbar use **data-icon** attribute to each anchor. By default, icons are placed at the top of the text.

## Example

Following example demonstrates the use of navbar icons in jQuery Mobile.

```
<!DOCTYPE html>
<head>
    <title>Navbar Icons</title>
    <meta name="viewport" content="width=device-width, initial-scale=1">
    <link rel="stylesheet"
    href="https://code.jquery.com/mobile/1.4.5/jquery.mobile-1.4.5.min.css">
    <script src="https://code.jquery.com/jquery-1.11.3.min.js"></script>
    <script src="https://code.jquery.com/mobile/1.4.5/jquery.mobile-
1.4.5.min.js"></script>
</head>
<body>
    <div data-role="navbar">
        <ul>
            <li><a href="#" data-icon="mail">Messages</a></li>
            <li><a href="#" data-icon="user" class="ui-btn-
active">Friends</a></li>
            <li><a href="#" data-icon="star">Notifications</a></li>
        </ul>
    </div>
</body>
</html>
```

## Output

Let's carry out the following steps to see how the above code works:

- Save the above html code as **navbar\_icons.html** file in your server root folder.
- Open this HTML file as [http://localhost/navbar\\_icons.html](http://localhost/navbar_icons.html) and the following output will be displayed.

- [Messages](#)
- [Friends](#)
- [Notifications](#)

## Navbars Icons Position

---

### Description

The navbar icons can be positioned on the navbar container instead of for every single link. Navbar icons can be positioned left, right, top or bottom of the text using ***data-iconpos*** attribute.

### Example

Following example demonstrates the use of navbar icons position in jQuery Mobile.

```
<!DOCTYPE html>
<head>
    <title>Navbar Icons Position</title>
    <meta name="viewport" content="width=device-width, initial-scale=1">
    <link rel="stylesheet"
    href="https://code.jquery.com/mobile/1.4.5/jquery.mobile-1.4.5.min.css">
    <script src="https://code.jquery.com/jquery-1.11.3.min.js"></script>
    <script src="https://code.jquery.com/mobile/1.4.5/jquery.mobile-
    1.4.5.min.js"></script>
</head>
<body>
    <p>Icons placed at bottom:</p>
    <div data-role="navbar" data-iconpos="bottom">
        <ul>
            <li><a href="#" data-icon="mail">Messages</a></li>
            <li><a href="#" data-icon="user" class="ui-btn-active">Friends</a></li>
            <li><a href="#" data-icon="star">Notifications</a></li>
        </ul>
    </div>
</body>
```

```

<p>Icons placed at top:</p>
<div data-role="navbar" data-iconpos="top">
    <ul>
        <li><a href="#" data-icon="mail">Messages</a></li>
        <li><a href="#" data-icon="user" class="ui-btn-active">Friends</a></li>
        <li><a href="#" data-icon="star">Notifications</a></li>
    </ul>
</div>
<p>Icons placed at left:</p>
<div data-role="navbar" data-iconpos="left">
    <ul>
        <li><a href="#" data-icon="mail">Messages</a></li>
        <li><a href="#" data-icon="user" class="ui-btn-active">Friends</a></li>
        <li><a href="#" data-icon="star">Notifications</a></li>
    </ul>
</div>
<p>Icons placed at right:</p>
<div data-role="navbar" data-iconpos="right">
    <ul>
        <li><a href="#" data-icon="mail">Messages</a></li>
        <li><a href="#" data-icon="user" class="ui-btn-active">Friends</a></li>
        <li><a href="#" data-icon="star">Notifications</a></li>
    </ul>
</div>
</body>
</html>

```

## Output

Let's carry out the following steps to see how the above code works:

- Save the above html code as **navbar\_icons\_position.html** file in your server root folder.
- Open this HTML file as [http://localhost/navbar\\_icons\\_position.html](http://localhost/navbar_icons_position.html) and the following output will be displayed.

Icons placed at bottom:

- [Messages](#)
- [Friends](#)
- [Notifications](#)

Icons placed at top:

- [Messages](#)
- [Friends](#)
- [Notifications](#)

Icons placed at left:

- [Messages](#)
- [Friends](#)
- [Notifications](#)

Icons placed at right:

- [Messages](#)
- [Friends](#)
- [Notifications](#)

## Navbars 3rd Party Icon Sets

---

### Description

You can add third party icons to your navbar. To achieve this, just add custom styles to link to the icons and place them in the navbar.

### Example

Following example demonstrates the use of navbars 3rd party icon in jQuery Mobile.

```
<!DOCTYPE html>
<html>
<head>
    <title>Navbars 3rd Party Icons</title>
    <meta name="viewport" content="width=device-width, initial-scale=1">
    <link rel="stylesheet"
    href="https://code.jquery.com/mobile/1.4.5/jquery.mobile-1.4.5.min.css">
    <script src="https://code.jquery.com/jquery-1.11.3.min.js"></script>
```

```

<script src="https://code.jquery.com/mobile/1.4.5/jquery.mobile-
1.4.5.min.js"></script>

<style>
    .nav-glyphish-example .ui-btn { padding-top: 40px !important; }
    .nav-glyphish-example .ui-btn:after { width: 30px!important; height:
30px!important; margin-left: -15px !important; }

    #home:after { background: url("/jquery_mobile/images/home.png") no-
repeat; }

    #notify:after { background:
url("/jquery_mobile/images/notifications.png") no-repeat; }
</style>

</head>

<body>

<div data-role="footer" class="nav-glyphish-example" data-theme="a">
    <div data-role="navbar" class="nav-glyphish-example" data-grid="a">
        <ul>
            <li><a href="#" id="home" data-icon="custom">Home</a></li>
            <li><a href="#" id="notify" data-icon="custom">Notifications</a></li>

        </ul>
    </div>
</div>
</body>
</html>

```

## Output

Let's carry out the following steps to see how the above code works:

- Save the above html code as **navbar\_3rd\_party\_icons.html** file in your server root folder.
- Open this HTML file as [http://localhost/navbar\\_3rd\\_party\\_icons.html](http://localhost/navbar_3rd_party_icons.html) and the following output will be displayed.

- [Home](#)
- [Notifications](#)

## Navbars Themes

---

### Description

Just like buttons, the theme swatch will be inherited by the navbars from their parent container. Whenever the navbar is positioned in the header or footer toolbar, the default toolbar swatch a will be inherited unless it is set in the markup.

You can add **swatch classes** to apply the body swatch (ui-body-a and ui-body-b). You can also use **data-theme** attribute to set the theme color for individual navbar items and specify a theme swatch.

The class ui-body adds the standard body padding.

### Example

Following example demonstrates the use of navbar themes in jQuery Mobile.

```
<!DOCTYPE html>
<html>
<head>
    <title>Navbars Themes</title>
    <meta name="viewport" content="width=device-width, initial-scale=1">
    <link rel="stylesheet"
    href="https://code.jquery.com/mobile/1.4.5/jquery.mobile-1.4.5.min.css">
    <script src="https://code.jquery.com/jquery-1.11.3.min.js"></script>
    <script src="https://code.jquery.com/mobile/1.4.5/jquery.mobile-
    1.4.5.min.js"></script>
</head>
<body>
    <div class="ui-body-a ui-body">
        <h3>Swatch "a"</h3>
        <div data-role="navbar">
```

```
<ul>
    <li><a href="#" data-icon="star">A</a></li>
    <li><a href="#" data-icon="gear">B</a></li>
    <li><a href="#" data-icon="grid">C</a></li>
    <li><a href="#" data-icon="arrow-l">D</a></li>
    <li><a href="#" data-icon="arrow-r">E</a></li>
</ul>
</div>
</div>
<div class="ui-body-b ui-body">
    <h3>Swatch "b"</h3>
    <div data-role="navbar">
        <ul>
            <li><a href="#" data-icon="star">A</a></li>
            <li><a href="#" data-icon="gear">B</a></li>
            <li><a href="#" data-icon="grid">C</a></li>
            <li><a href="#" data-icon="arrow-l">D</a></li>
            <li><a href="#" data-icon="arrow-r">E</a></li>
        </ul>
    </div>
</div>
<h3><i>data-theme</i> attribute</h3>
<div data-role="footer">
    <div data-role="navbar">
        <ul>
            <li><a href="#" data-icon="grid" data-theme="a">A</a></li>
            <li><a href="#" data-icon="grid" data-theme="b">B</a></li>
        </ul>
    </div>
</div>
</body>
</html>
```

## Output

Let's carry out the following steps to see how the above code works:

- Save the above html code as **navbar\_themes.html** file in your server root folder.
- Open this HTML file as [http://localhost/navbar\\_themes.html](http://localhost/navbar_themes.html) and the following output will be displayed.

### Swatch "a"

- [A](#)
- [B](#)
- [C](#)
- [D](#)
- [E](#)

### Swatch "b"

- [A](#)
- [B](#)
- [C](#)
- [D](#)
- [E](#)

### *data-theme* attribute

- [A](#)
- [B](#)

## Navbars with Button Elements

---

### Description

Instead of using links inside the navbar, you can also use button elements.

### Example

Following example demonstrates the use of navbar with button in jQuery Mobile.

```
<!DOCTYPE html>
<html>
<head>
  <title>Navbars with button elements</title>
```

```

<meta name="viewport" content="width=device-width, initial-scale=1">
<link rel="stylesheet"
href="https://code.jquery.com/mobile/1.4.5/jquery.mobile-1.4.5.min.css">
<script src="https://code.jquery.com/jquery-1.11.3.min.js"></script>
<script src="https://code.jquery.com/mobile/1.4.5/jquery.mobile-
1.4.5.min.js"></script>
</head>
<body>
<div data-role="navbar" data-iconpos="left">
<ul>
<li><button data-icon="star">Button One</button></li>
<li><button data-icon="arrow-l" data-theme="b">Button
Two</button></li>
<li><button data-icon="arrow-r">Button Three</button></li>
</ul>
</div>
</body>
</html>

```

## Output

Let's carry out the following steps to see how the above code works:

- Save the above html code as **navbar\_with\_button.html** file in your server root folder.
- Open this HTML file as [http://localhost/navbar\\_with\\_button.html](http://localhost/navbar_with_button.html) and the following output will be displayed.



## Panels Widget

---

### Description

Panels are used to display the DOM components in the box using the ***data-role="panel"*** attribute.

Following table lists some of the panel types, which are supported by the jQuery Mobile.

| Sr. No. | Type & Description  |
|---------|---|
| 1       | <b><u>Panels</u></b><br>Basic panel which moves from the left or the right side of the screen to display the content.           |
| 2       | <b><u>Closing Panels</u></b><br>You can close the panel using Esc key or by clicking outside the panel.                         |
| 3       | <b><u>External Panels</u></b><br>External panels are used to display the content outside the page.                              |
| 4       | <b><u>Positioning Panels</u></b><br>Specifies the position of panel in the page.  |
| 5       | <b><u>Panel Responsive</u></b><br>You can make the panel responsive and allows the panel menu and the page to be used together. |
| 6       | <b><u>Panel Styling</u></b><br>You can use styles for the panels.   |
| 7       | <b><u>Open panel on swipe</u></b><br>You can open the panel with a swipe.   |
| 8       | <b><u>External and Internal Panel</u></b><br>Displays the internal and external panels.   |

## Panels

---

### Description

Basic panel which moves from the left or the right side of the screen to display the content using ***data-role="panel"*** attribute.

## Example

Following example describes the use of panel in the jQuery Mobile framework.

```
<!DOCTYPE html>
<head>
<title>Panel</title>
<meta name="viewport" content="width=device-width, initial-scale=1">
<link rel="stylesheet"
href="https://code.jquery.com/mobile/1.4.5/jquery.mobile-1.4.5.min.css">
<script src="https://code.jquery.com/jquery-1.11.3.min.js"></script>
<script src="https://code.jquery.com/mobile/1.4.5/jquery.mobile-
1.4.5.min.js"></script>
</head>
<body>
<div data-role="page" id="page1">
<div data-role="panel" id="panel1">
<h2>This is Panel Header</h2>
<p>You can close this panel by pressing the Esc key or by swiping.</p>
</div>
<div data-role="header">
<h2>Header</h2>
</div>
<div data-role="main" class="ui-content">
<p>Click on the button to open Panel.</p>
<a href="#panel1" class="ui-btn ui-btn-inline">Open Panel</a>
</div>
<div data-role="footer">
<h2>Footer</h2>
</div>
</div>
</body>
</html>
```

## Output

Let's carry out the following steps to see how the above code works:

- Save the above html code as **jqm\_panels.html** file in your server root folder.
- Open this HTML file as [http://localhost/jqm\\_panels.html](http://localhost/jqm_panels.html) and the following output will be displayed.

## This is Panel Header

You can close this panel by pressing the Esc key or by swiping.

### Header

Click on the button to open Panel.

[Open Panel](#)

### Footer

## Panel Markup

You can write the panel content inside a page and it is sibling to the header, the content and the footer elements. You cannot write the panel markup outside the page. It will have the following format.

```
<div data-role="page">
<div data-role="panel" id="panel1">
    <!-- panel content -->
</div><!-- end of panel -->
<!-- header -->
<!-- content -->
<!-- footer -->
</div><!-- end of page -->
```

## Dynamic Content

You can dynamically add the content to a panel or hide the content when the panel is open, using the **updatelayout** event on the panel as shown in the following format.

```
$( "#panel1" ).trigger( "updatelayout" );
```

It will check for the height of panel contents and if the height of the panel exceeds, then it will set the height of the content by using ***min-height*** attribute and the position of the panel by using ***data-position-fixed="true"*** attribute.

## Opening a Panel

You can open the panel by setting ***href*** attribute to the id of the panel, which ties the link to the panel. It will open the panel when you click on the link and close the panel by clicking it again.

## Closing a Panel

You can close the panel by clicking outside the panel, swiping left or right, or pressing Esc key. The swiping effect can be turned off using the ***data-swipe-close="false"*** attribute. Sometimes, you can close the panel by clicking outside the panel; so that you can avoid this action by adding the ***data-dismissible="false"*** to the panel. It's also possible to close the panel by adding the ***data-rel="close"*** attribute to the panel.

## Panel Animations

You can use animations on the panel if your browser provides support for 3D transforms. Use the ***translate3d(x,y,z)*** CSS transforms to animate the panel. If you don't want to use animations, then use the ***data-animate="false"*** attribute to the panel container.

## Panel Positioning

It specifies the position of panel in the page and displays the panel with the ***position:absolute*** CSS property. Use the ***position:fixed*** attribute in the panel to display the content and it doesn't matter how much you scroll the page using ***data-position-fixed="true"*** attribute to the panel. If the content is more which cannot fit within the page area, then the framework will display the content without fixed positioning.

## Styling Panels

The jQuery Mobile framework provides styles for the panels. By default, panels will be in simple styles but you can customize them according to the user criteria. You can enclose the content in a div using the ***ui-panel-inner*** class which includes 15 pixels of padding. You can set the theme background for the panels using different types of themes. The default theme for panel is "c" and set the theme for panel using the ***data-theme*** attribute. If you don't want to set the theme, then use the ***data-theme="none"*** attribute to use your own classes to style the panel.

## Making the Panel Responsive

The jQuery Mobile framework makes the panel responsive and allows the panel menu and page to be used together. You can use specific breakpoint to a page or use the ***class = "ui-responsive-panel"*** breakpoint preset to the page container.

## Closing Panel

---

### Description

You can close the panel using Esc key or by clicking outside the panel using ***data-rel="close"*** attribute.

### Example

Following example describes the use of closing panel in the jQuery Mobile framework.

```
<!DOCTYPE html>
<head>
<title>Closing Panel</title>
<meta name="viewport" content="width=device-width, initial-scale=1">
<link rel="stylesheet"
href="https://code.jquery.com/mobile/1.4.5/jquery.mobile-1.4.5.min.css">
<script src="https://code.jquery.com/jquery-1.11.3.min.js"></script>
<script src="https://code.jquery.com/mobile/1.4.5/jquery.mobile-
1.4.5.min.js"></script>
</head>
<body>
<div data-role="page" id="page1">
<div data-role="panel" id="panel1">
<h2>This is Panel Header</h2>
<p>You can close this panel by pressing the Esc key or by swiping or
clicking outside the panel.</p>
<a href="#page1" data-rel="close" class="ui-btn ui-btn-inline ui-icon-
delete ui-btn-icon-left">Close panel</a>
</div>
<div data-role="header">
<h2>Header</h2>
</div>
<div data-role="main" class="ui-content">
<p>Click on the button to open Panel.</p>
<a href="#panel1" class="ui-btn ui-btn-inline">Open Panel</a>
</div>
<div data-role="footer">
<h2>Footer</h2>
</div>
</div>
```

```
</body>
</html>
```

## Output

Let's carry out the following steps to see how the above code works:

- Save the above html code as **jqm\_closing\_panels.html** file in your server root folder.
- Open this HTML file as [http://localhost/jqm\\_closing\\_panels.html](http://localhost/jqm_closing_panels.html) and the following output will be displayed.

## This is Panel Header

You can close this panel by pressing the Esc key or by swiping or clicking outside the panel.

[Close panel](#)

## Header

Click on the button to open Panel.

[Open Panel](#)

## Footer

## External Panel

---

### Description

External panels are used to display the content outside the page by setting the **id** to href of the panel.

### Example

Following example describes the use of external panel in the jQuery Mobile framework.

```
<!DOCTYPE html>
<head>
<title>External Panel</title>
<meta name="viewport" content="width=device-width, initial-scale=1">
```

```

<link rel="stylesheet"
href="https://code.jquery.com/mobile/1.4.5/jquery.mobile-1.4.5.min.css">
<script src="https://code.jquery.com/jquery-1.11.3.min.js"></script>
<script src="https://code.jquery.com/mobile/1.4.5/jquery.mobile-
1.4.5.min.js"></script>
</head>
<body>
<div data-role="page" id="page1">
  <div data-role="panel" id="overlay_panel" data-display="overlay">
    <h2>This is Overlay Panel</h2>
    <p>You can close this panel by pressing the Esc key or by swiping or
clicking outside the panel.</p>
    <a href="#page1" data-rel="close" class="ui-btn ui-btn-inline ui-btn-a ui-
icon-delete ui-btn-icon-left">Close Panel</a>
  </div>
  <div data-role="panel" id="reveal_panel" data-display="reveal">
    <h2>Reveal Panel</h2>
    <p>You can close this panel by pressing the Esc key or by swiping or
clicking outside the panel.</p>
    <a href="#page1" data-rel="close" class="ui-btn ui-btn-inline ui-btn-a ui-
icon-delete ui-btn-icon-left">Close Panel</a>
  </div>
  <div data-role="panel" id="push_panel" data-display="push">
    <h2>Push Panel</h2>
    <p>You can close this panel by pressing the Esc key or by swiping or
clicking outside the panel.</p>
    <a href="#page1" data-rel="close" class="ui-btn ui-btn-inline ui-btn-a ui-
icon-delete ui-btn-icon-left">Close Panel</a>
  </div>
  <div data-role="header">
    <h2>Header</h2>
  </div>
  <div data-role="main" class="ui-content">
    <p>Click the below buttons to open the Panel with different display
modes.</p>
    <a href="#overlay_panel" class="ui-btn ui-btn-inline ui-shadow">This is
Overlay Panel</a>
    <a href="#reveal_panel" class="ui-btn ui-btn-inline ui-shadow">This is
Reveal Panel</a>
    <a href="#push_panel" class="ui-btn ui-btn-inline ui-shadow">This is Push
Panel</a>
  </div>
</div>

```

```
</div>
<div data-role="footer">
    <h2>Footer</h2>
</div>
</div>
</body>
</html>
```

## Output

Let's carry out the following steps to see how the above code works:

- Save the above html code as **jqm\_external\_panels.html file** in your server root folder.
- Open this HTML file as [http://localhost/jqm\\_external\\_panels.html](http://localhost/jqm_external_panels.html) and the following output will be displayed.

## This is Overlay Panel

You can close this panel by pressing the Esc key or by swiping or clicking outside the panel.

[Close Panel](#)

## Reveal Panel

You can close this panel by pressing the Esc key or by swiping or clicking outside the panel.

[Close Panel](#)

## Push Panel

You can close this panel by pressing the Esc key or by swiping or clicking outside the panel.

[Close Panel](#)

## Header

Click the below buttons to open the Panel with different display modes.

[This is Overlay Panel](#) [This is Reveal Panel](#) [This is Push Panel](#)

## Footer

## Fixed Panel Positioning

---

### Description

You can specify the position of panel (right or left) by using the **data-position** attribute in the page.

### Example

Following example describes the use of fixed panel positioning in the jQuery Mobile framework.

```
<!DOCTYPE html>
<head>
<title>Fixed Positioning Panel</title>
```

```

<meta name="viewport" content="width=device-width, initial-scale=1">
<link rel="stylesheet"
href="https://code.jquery.com/mobile/1.4.5/jquery.mobile-1.4.5.min.css">
<script src="https://code.jquery.com/jquery-1.11.3.min.js"></script>
<script src="https://code.jquery.com/mobile/1.4.5/jquery.mobile-
1.4.5.min.js"></script>
</head>
<body>
<div data-role="page" id="page1">
    <div data-role="panel" id="panel1" data-position="left">
        <h2>My Panel</h2>
        <p>You can close this panel by pressing the Esc key or by swiping or
clicking outside the panel.</p>
        <a href="#page1" data-rel="close" class="ui-btn ui-btn-inline ui-btn-a
ui-icon-delete ui-btn-icon-left">Close panel</a>
    </div>
    <div data-role="header">
        <h2>Header</h2>
    </div>
    <div data-role="main" class="ui-content">
        <p>Click on the button to open the Panel.</p>
        <a href="#panel1" class="ui-btn ui-btn-inline">Open Panel</a>
    </div>
    <div data-role="footer">
        <h2>Footer</h2>
    </div>
</div>
</body>
</html>

```

## Output

Let's carry out the following steps to see how the above code works:

- Save the above html code as **jqm\_fixed\_positioning.html** file in your server root folder.
- Open this HTML file as [http://localhost/jqm\\_fixed\\_positioning.html](http://localhost/jqm_fixed_positioning.html) and the following output will be displayed.

## My Panel

You can close this panel by pressing the Esc key or by swiping or clicking outside the panel.

[Close panel](#)

### Header

Click on the button to open the Panel.

[Open Panel](#)

### Footer

## Panel Responsive

---

### Description

You can make the panel responsive and allow the panel menu and the page to be used together using the ***ui-responsive-panel*** class.

### Example

Following example describes the use of panel responsive in the jQuery Mobile framework.

```
<!DOCTYPE html>
<head>
<title>Panel Responsive</title>
<meta name="viewport" content="width=device-width, initial-scale=1">
<link rel="stylesheet"
href="https://code.jquery.com/mobile/1.4.5/jquery.mobile-1.4.5.min.css">
<script src="https://code.jquery.com/jquery-1.11.3.min.js"></script>
<script src="https://code.jquery.com/mobile/1.4.5/jquery.mobile-
1.4.5.min.js"></script>
</head>
<body>
<div data-role="page" class="jqm-demos ui-responsive-panel"
id="responsive_page">
<div data-role="header">
<h2>Header</h2>
```

```

<a href="#panel1" data-icon="home" data-iconpos="notext">Right Panel</a>
<a href="#panel2" data-icon="grid" data-iconpos="notext">Left Panel</a>
</div>
<div role="main" class="ui-content jqm-content jqm-fullwidth">
    <h2>Panel Responsive</h2>
    <p>It has two buttons in the header bar namely the left panel has the push display mode and the right panel reveal display mode. It allows both the panel menu and page to be used together when more space is available. It makes the panel responsive and allows panel menu and page to be used together. You can use custom specific breakpoint or use the breakpoint preset by adding the <code>class="ui-responsive-panel"</code> to the page container.</p>
</div>
<div data-role="panel" data-display="push" data-theme="b" id="panel1">
    <ul data-role="listview">
        <li data-icon="delete"><a href="#" data-rel="close">Close Menu</a></li>
        <li><a href="#responsive_page1">Java</a></li>
        <li><a href="#responsive_page1">Ajax</a></li>
        <li><a href="#responsive_page1">HTML</a></li>
        <li><a href="#responsive_page1">CSS</a></li>
        <li><a href="#responsive_page1">PHP</a></li>
        <li><a href="#responsive_page1">Bootstrap</a></li>
        <li><a href="#responsive_page1">JavaScript</a></li>
    </ul>
</div>
<div data-role="panel" data-position="right" data-display="overlay" data-theme="b" id="panel2">
    <ul data-role="listview">
        <li data-icon="delete"><a href="#" data-rel="close">Close Menu</a></li>
        <li><a href="#responsive_page1">Java</a></li>
        <li><a href="#responsive_page1">Ajax</a></li>
        <li><a href="#responsive_page1">HTML</a></li>
        <li><a href="#responsive_page1">CSS</a></li>
        <li><a href="#responsive_page1">PHP</a></li>
        <li><a href="#responsive_page1">Bootstrap</a></li>
        <li><a href="#responsive_page1">JavaScript</a></li>
    </ul>
</div>

```

```

<div data-role="footer">
    <h2>Footer</h2>
</div>
</div>
</body>
</html>

```

## Output

Let's carry out the following steps to see how the above code works:

- Save the above html code as **jqm\_panel\_responsive.html** file in your server root folder.
- Open this HTML file as [http://localhost/jqm\\_panel\\_responsive.html](http://localhost/jqm_panel_responsive.html) and the following output will be displayed.

## Header

[Right Panel](#) [Left Panel](#)

## Panel Responsive

It has two buttons in the header bar namely the left panel has the push display mode and the right panel reveal display mode. It allows both the panel menu and page to be used together when more space is available. It makes the panel responsive and allows panel menu and page to be used together. You can use custom specific breakpoint or use the breakpoint preset by adding the `class="ui-responsive-panel"` to the page container.

- [Close Menu](#)
- [Java](#)
- [Ajax](#)
- [HTML](#)
- [CSS](#)
- [PHP](#)
- [Bootstrap](#)
- [JavaScript](#)
  
- [Close Menu](#)
- [Java](#)
- [Ajax](#)
- [HTML](#)
- [CSS](#)
- [PHP](#)
- [Bootstrap](#)
- [JavaScript](#)

## Footer

## Panel Styling

### Description

You can use styles for the panels.

### Example

Following example describes the use of panel styling in the jQuery Mobile framework.

```
<!DOCTYPE html>
<head>
<title>Panel Styling</title>
<meta name="viewport" content="width=device-width, initial-scale=1">
<link rel="stylesheet"
href="https://code.jquery.com/mobile/1.4.5/jquery.mobile-1.4.5.min.css">
<script src="https://code.jquery.com/jquery-1.11.3.min.js"></script>
<script src="https://code.jquery.com/mobile/1.4.5/jquery.mobile-
1.4.5.min.js"></script>
<style>
h2{
color:#BF00FF;
}
.article {
background-color:rgb(192,192,192);
}
</style>
</head>
<body>
<div data-role="page" id="my_page">
<div data-role="header">
<h2>Header</h2>
<a href="#left-panel" data-icon="grid" data-iconpos="notext">Right
Panel</a>
</div>
<div role="main" class="ui-content">
<div class="article">
<h2>jQuery Mobile</h2>
<p>jQuery Mobile is a touch-optimized web framework for creating mobile
web applications. jQuery Mobile works on all popular smartphones and tablets.
jQuery Mobile is built on top of the jQuery library, which makes it easy to
learn if you already know jQuery.</p>

```

```

    </div>
</div>
<div data-role="panel" id="left-panel">
    <ul data-role="listview">
        <li data-icon="home"><a href="#">India</a></li>
        <li data-icon="lock"><a href="#">Srilanka</a></li>
        <li data-role="audio"><a href="#">New Zealand</a></li>
    </ul>
</div>
</div>
</body>
</html>

```

## Output

Let's carry out the following steps to see how the above code works:

- Save the above html code as **jqm\_panels\_styling.html** file in your server root folder.
- Open this HTML file as [http://localhost/jqm\\_panels\\_styling.html](http://localhost/jqm_panels_styling.html) and the following output will be displayed.

## Header

[Right Panel](#)

## jQuery Mobile

jQuery Mobile is a touch-optimized web framework for creating mobile web applications. jQuery Mobile works on all popular smartphones and tablets. jQuery Mobile is built on top of the jQuery library, which makes it easy to learn if you already know jQuery.

- [India](#)
- [Srilanka](#)
- [New Zealand](#)

## Panel Swipe

---

### Description

You can open the panel with a swipe by setting the ***href*** to the ***id*** of the panel.

### Example

Following example describes the use of panel swipe in the jQuery Mobile framework.

```
<!DOCTYPE html>
<head>
<title>Panel Swipe</title>
<meta name="viewport" content="width=device-width, initial-scale=1">
<link rel="stylesheet"
href="https://code.jquery.com/mobile/1.4.5/jquery.mobile-1.4.5.min.css">
<script src="https://code.jquery.com/jquery-1.11.3.min.js"></script>
<script src="https://code.jquery.com/mobile/1.4.5/jquery.mobile-
1.4.5.min.js"></script>
<script>
$( document ).on( "pagecreate", "#mypage", function() {
    $( document ).on( "swipeleft swiperight", "#mypage", function( e ) {
        if ( $( ".ui-page-active" ).jqmData( "panel" ) !== "open" ) {
            if ( e.type === "swipeleft" ) {
                $( "#right-panel" ).panel( "open" );
            } else if ( e.type === "swiperight" ) {
                $( "#left-panel" ).panel( "open" );
            }
        }
    });
});
</script>
</head>
<body>
<div data-role="page" id="mypage">
    <div data-role="header" data-theme="b">
        <h2>Header</h2>
        <a href="#left-panel" data-icon="carat-r" data-iconpos="notext"
class="ui-nodisc-icon">Open left panel</a>
        <a href="#right-panel" data-icon="carat-l" data-iconpos="notext"
class="ui-nodisc-icon">Open right panel</a>
    </div>
    <div data-role="content">
        Content goes here
    </div>
</div>

```

```

</div>
<div role="main" class="ui-content">
    <p>Swipe the panel left or right.</p>
</div>
<div data-role="panel" id="left-panel" data-theme="b">
    <p>This is left panel.</p>
    <a href="#" data-rel="close" class="ui-btn ui-btn-inline ui-icon-delete ui-btn-icon-left">Close Panel</a>
</div>
<div data-role="panel" id="right-panel" data-display="push" data-position="right" data-theme="b">
    <p>This is right panel.</p>
    <a href="#" data-rel="close" class="ui-btn ui-btn-inline ui-icon-delete ui-btn-icon-right">Close Panel</a>
</div>
</div>
</body>
</html>

```

## Output

Let's carry out the following steps to see how the above code works:

- Save the above html code as **jqm\_panels\_on\_swipe.html** file in your server root folder.
- Open this HTML file as [http://localhost/jqm\\_panels\\_on\\_swipe.html](http://localhost/jqm_panels_on_swipe.html) and the following output will be displayed.

## Header

[Open left panel](#) [Open right panel](#)

Swipe the panel left or right.

This is left panel.

[Close Panel](#)

This is right panel.

[Close Panel](#)

## Internal and External Panels

---

### Description

You can display the internal and external panels by giving the panels path in the ***href*** attribute.

### Example

Following example describes the use of internal *and* external panels in the jQuery Mobile framework.

```
<!DOCTYPE html>
<head>
<title>External and Internal Panels</title>
<meta name="viewport" content="width=device-width, initial-scale=1">
<link rel="stylesheet"
href="https://code.jquery.com/mobile/1.4.5/jquery.mobile-1.4.5.min.css">
<script src="https://code.jquery.com/jquery-1.11.3.min.js"></script>
<script src="https://code.jquery.com/mobile/1.4.5/jquery.mobile-
1.4.5.min.js"></script>
<script>
$( document ).on( "pagecreate", function() {
    $( "body > [data-role='panel']" ).panel();
    $( "body > [data-role='panel'] [data-role='listview']" ).listview();
});
$( document ).one( "pageshow", function() {
    $( "body > [data-role='header']" ).toolbar();
    $( "body > [data-role='header'] [data-role='navbar']" ).navbar();
});
</script>
</head>
<body>
    <div data-role="header" data-position="fixed" data-theme="a">
        <h2>Header</h2>
    </div>
    <div data-role="page">
        <div role="main" class="ui-content">
            <h2>This is first page.</h2>
            <a href="#page_one" class="ui-btn ui-btn-inline">Click here to open
right panel</a>
    
```

```

</div>

<div data-role="panel" id="page_one" data-position="right" data-
display="overlay" data-theme="b">
    <ul data-role="listview">
        <li data-icon="delete"><a href="#" data-rel="close">Close</a></li>
        <li>New Zealand</li>
        <li>England</li>
        <li>South Africa</li>
    </ul>
</div>
<div data-role="footer" data-position="fixed">
    <div data-role="navbar">
        <ul>
            <li><a href="/jquery_mobile/src/jqm_external_internal_panel.html"
class="ui-btn-active">Page One</a></li>
            <li><a href="/jquery_mobile/src/page2.html">Page Two</a></li>
        </ul>
    </div>
</div>
</body>
</html>

```

## Output

Let's carry out the following steps to see how the above code works:

- Save the above html code as **jqm\_external\_internal\_panel.html** file in your server root folder.
- Open this HTML file as [http://localhost/jqm\\_external\\_internal\\_panel.html](http://localhost/jqm_external_internal_panel.html) and the following output will be displayed.

## Header

### This is first page.

[Click here to open right panel](#)

- [Close](#)
- [New Zealand](#)
- [England](#)
- [South Africa](#)
  
- [Page One](#)
- [Page Two](#)

## Popup Widget

---

### Description

Popup is a user interface that appears within a small window to display text, images, and other content.

Following table lists some of the popup types, which are supported by the jQuery Mobile.

| Sr. No. | Type & Description   |
|---------|--|
| 1       | <b><a href="#">Popup Basics</a></b><br>Displays the content within a small window.   |
| 2       | <b><a href="#">Popup Alignment</a></b><br>Sets the alignment for the popup in a page.  |
| 3       | <b><a href="#">Dynamic Popup</a></b><br>Creates the popup dynamically in which popup gets the right size and position.                                   |
| 4       | <b><a href="#">Popups iframes</a></b><br>Uses the iframe inside a popup to use videos and maps.  |
| 5       | <b><a href="#">Scaling Images</a></b><br>Scales the images in landscape and portrait format.   |
| 6       | <b><a href="#">Popup Across Multiple Pages</a></b><br>You can use the same popup on multiple pages by specifying the popup as child of the body element. |

## Basic Popup

---

### Description

It displays the content within a small window.

Following table lists some of the basic popup types, which are supported by the jQuery Mobile.

| Sr. No. | Type & Description   |
|---------|--|
| 1       | <b>Basics of Popup</b><br>Popup looks like a small window which displays the text, photos and other content.             |
| 2       | <b>Tooltip</b><br>Displays the small popup box when you hover the mouse on the element.                                  |
| 3       | <b>Closing Popups</b><br>You can close the popups by clicking outside the popup box or by pressing the Esc key.          |
| 4       | <b>Photo Lightbox</b><br>You can use lightbox for keeping the images in the popup.                                       |
| 5       | <b>Popup Menu</b><br>Creates a menu by using listview in the popup.  |
| 6       | <b>Popup Form</b><br>Defines the form in the popup.  |
| 7       | <b>Popup Dialog</b><br>Places the dialog box inside the popup container.   |
| 8       | <b>Popup with Padding</b><br>Specifies the padding for the text in a popup.  |
| 9       | <b>Popup Position</b><br>Specifies the position for popup box.   |
| 10      | <b>Popup Transitions</b><br>You can use variety of effects which specifies the transition from one page to another page. |
| 11      | <b>Popup Theme</b>   |

|    |   |
|----|---|
|    | Applies the different types of theme for the popup.   |
| 12 | <b><u>Popup Arrow</u></b><br>Displays the popup with an arrow.  |
| 13 | <b><u>Pre-rendered Popup Markup</u></b><br>You can use pre-rendered markup in the popup to save the startup time. |

## Basics of Popup

---

### Description

Popup looks like a small window which is used to display the text, photos, and other content.

### Example

Following example demonstrates the use of popup in the jQuery Mobile Framework.

```
<!DOCTYPE html>
<head>
<title>Popup Basics</title>
<meta name="viewport" content="width=device-width, initial-scale=1">
<link rel="stylesheet"
href="https://code.jquery.com/mobile/1.4.5/jquery.mobile-1.4.5.min.css">
<script src="https://code.jquery.com/jquery-1.11.3.min.js"></script>
<script src="https://code.jquery.com/mobile/1.4.5/jquery.mobile-
1.4.5.min.js"></script>
</head>
<body>
<div data-role="page">
  <div data-role="header">
    <h2>Header</h2>
  </div>
  <div data-role="main" class="ui-content">
    <a href="#popup_example" data-rel="popup" class="ui-btn ui-btn-
inline">Click here to see popup</a>
    <div data-role="popup" id="popup_example">
      <p>Welcome to TutorialsPoint.....</p>
    </div>
  </div>
</div>
```

```
<div data-role="footer">
    <h2>Footer</h2>
</div>
</div>
</body>
</html>
```

## Output

Let's carry out the following steps to see how the above code works:

- Save the above html code as **popup\_basics.html** file in your server root folder.
- Open this HTML file as [http://localhost/popup\\_basics.html](http://localhost/popup_basics.html) and the following output will be displayed.

## Header

[Click here to see popup](#)

Welcome to Tutorialspoint.....

## Footer

## Popup Tooltip

---

### Description

Displays the small popup box when you hover the mouse on the element using **data-rel="popup"** and **data-transition="pop"** attributes **ui-content** class.

### Example

Following example demonstrates the use of popup tooltip in the jQuery Mobile Framework.

```
<!DOCTYPE html>
<head>
<title>Popup Tooltip</title>
<meta name="viewport" content="width=device-width, initial-scale=1">
<link rel="stylesheet"
href="https://code.jquery.com/mobile/1.4.5/jquery.mobile-1.4.5.min.css">
```

```

<script src="https://code.jquery.com/jquery-1.11.3.min.js"></script>
<script src="https://code.jquery.com/mobile/1.4.5/jquery.mobile-
1.4.5.min.js"></script>
</head>
<body>
<div data-role="page">
    <div data-role="header">
        <h2>Header</h2>
    </div>
    <div data-role="main" class="ui-content">
        <p>This is example of tooltip. <a href="#popup_tooltip" data-rel="popup"
data-transition="pop" class="ui-btn ui-alt-icon ui-nodisc-icon ui-btn-inline
ui-icon-info ui-btn-icon-notext" title="This is Tooltip">More Info</a></p>
        <div data-role="popup" id="popup_tooltip" class="ui-content" data-
theme="a">
            <p>Welcome to TutorialsPoint.....</p>
        </div>
    </div>
    <div data-role="footer">
        <h2>Footer</h2>
    </div>
</div>
</body>
</html>

```

## Output

Let's carry out the following steps to see how the above code works:

- Save the above html code as **jqm\_tooltip.html** file in your server root folder.
- Open this HTML file as [http://localhost/jqm\\_tooltip.html](http://localhost/jqm_tooltip.html) and the following output will be displayed.

## Header

This is example of tooltip. [More Info](#)

Welcome to Tutorialspoint.....

## Footer

## Closing Popups

---

### Description

You can close the popups by clicking outside the popup box or by pressing the Esc key. You can stop the closable by clicking outside the popup box using the ***data-dismissible="false"*** attribute.

### Example

Following example demonstrates the use of closing popups in the jQuery Mobile Framework.

```
<!DOCTYPE html>
<head>
<title>Closing Popups</title>
<meta name="viewport" content="width=device-width, initial-scale=1">
<link rel="stylesheet"
href="https://code.jquery.com/mobile/1.4.5/jquery.mobile-1.4.5.min.css">
<script src="https://code.jquery.com/jquery-1.11.3.min.js"></script>
<script src="https://code.jquery.com/mobile/1.4.5/jquery.mobile-
1.4.5.min.js"></script>
</head>
<body>
<div data-role="page">
<div data-role="header">
<h2>Header</h2>
</div>
<a href="#popup_right" data-rel="popup" class="ui-btn ui-btn-inline">Right
close button</a><br>
<a href="#popup_left" data-rel="popup" class="ui-btn ui-btn-inline">Left
close button</a><br>
```

```

<a href="#popup_undismissible" data-rel="popup" class="ui-btn ui-btn-inline">Undismissible Popup</a>

<div data-role="popup" id="popup_right" class="ui-content">
    <a href="#" data-rel="back" class="ui-btn ui-btn-a ui-icon-delete ui-btn-icon-notext ui-btn-right">Close</a>
    <p>Welcome to TutorialsPoint.....</p>
</div>

<div data-role="popup" id="popup_left" class="ui-content">
    <a href="#" data-rel="back" class="ui-btn ui-btn-a ui-icon-delete ui-btn-icon-notext ui-btn-left">Close</a>
    <p>Welcome to TutorialsPoint.....</p>
</div>

<div data-role="popup" id="popup_undismissible" class="ui-content" data-dismissible="false">
    <a href="#" data-rel="back" class="ui-btn ui-btn-a ui-icon-delete ui-btn-icon-notext ui-btn-left">Close</a>
    <p>You cannot close the popup by clicking outside. You can close this by clicking at top left corner.</p>
</div>
<div data-role="footer">
    <h2>Footer</h2>
</div>
</div>
</body>
</html>

```

## Output

Let's carry out the following steps to see how the above code works:

- Save the above html code as **jqm\_closing\_popups.html** file in your server root folder.
- Open this HTML file as [http://localhost/jqm\\_closing\\_popups.html](http://localhost/jqm_closing_popups.html) and the following output will be displayed.

## Header

[Right close button](#)

[Left close button](#)

[Undismissible Popup](#)

[Close](#)

Welcome to Tutorialspoint.....

[Close](#)

Welcome to Tutorialspoint.....

[Close](#)

You cannot close the popup by clicking outside. You can close this by clicking at top left corner.

## Footer

## Photo Lightbox

---

### Description

You can use the lightbox for keeping the images in the popup. You can enlarge the image by clicking it.

### Example

Following example demonstrates the use of photo lightbox popup in the jQuery Mobile Framework.

```
<!DOCTYPE html>
<head>
<title>Photo Lightbox</title>
<meta name="viewport" content="width=device-width, initial-scale=1">
<link rel="stylesheet"
href="https://code.jquery.com/mobile/1.4.5/jquery.mobile-1.4.5.min.css">
<script src="https://code.jquery.com/jquery-1.11.3.min.js"></script>
<script src="https://code.jquery.com/mobile/1.4.5/jquery.mobile-
1.4.5.min.js"></script>
</head>
<body>
```

```

<div data-role="page">
    <div data-role="header">
        <h2>Header</h2>
    </div>
    <div id="page1" data-role="main" class="ui-content">
        <p>You can enlarge the image clicking on it...</p>
        <a href="#popup_img" data-rel="popup" data-position-to="window">
            </a>

            <div data-role="popup" id="popup_img">
                <p>It's an amazing view...</p>
                <a href="#page1" class="ui-btn-a ui-icon-delete ui-btn-icon-notext ui-
                btn-right">Close</a>
            </div>
        </div>
        <div data-role="footer">
            <h2>Footer</h2>
        </div>
    </div>
</body>
</html>

```

## Output

Let's carry out the following steps to see how the above code works:

- Save the above html code as **jqm\_photo\_lightbox.html** file in your server root folder.
- Open this HTML file as [http://localhost/jqm\\_photo\\_lightbox.html](http://localhost/jqm_photo_lightbox.html) and the following output will be displayed.

## Header

You can enlarge the image clicking on it...



It's an amazing view...

[Close](#)



## Footer

## Popup Menu

---

### Description

Create the menu using ***data-role="listview"*** attribute in the popup container.

### Example

Following example demonstrates the use of popup menu in the jQuery Mobile Framework.

```
<!DOCTYPE html>
<head>
<title>Popup Menu</title>
<meta name="viewport" content="width=device-width, initial-scale=1">
<link rel="stylesheet"
href="https://code.jquery.com/mobile/1.4.5/jquery.mobile-1.4.5.min.css">
<script src="https://code.jquery.com/jquery-1.11.3.min.js"></script>
<script src="https://code.jquery.com/mobile/1.4.5/jquery.mobile-
1.4.5.min.js"></script>
</head>
<body>
<div data-role="page">
    <div data-role="header">
        <h2>Header</h2>
    </div>
    <div id="page1" data-role="main" class="ui-content">
        <a href="#popup_menu" data-rel="popup" data-transition="slidedown"
class="ui-btn ui-btn-inline ui-icon-home ui-btn-icon-left ui-btn-a">Click to
see menu items</a>
        <div data-role="popup" id="popup_menu" data-theme="a">
            <ul data-role="listview">
                <li data-role="list-divider">Select your choice</li>
                <li><a href="#">India</a></li>
                <li><a href="#">Srilanka</a></li>
                <li><a href="#">Australia</a></li>
                <li><a href="#">England</a></li>
            </ul>
        </div>
    </div>
    <div data-role="footer">
        <h2>Footer</h2>
    </div>
</div>
```

```

</div>
</div>
</body>
</html>
```

## Output

Let's carry out the following steps to see how the above code works:

- Save the above html code as **jqm\_popup\_menu.html** file in your server root folder.
- Open this HTML file as [http://localhost/jqm\\_popup\\_menu.html](http://localhost/jqm_popup_menu.html) and the following output will be displayed.

## Header

[Click to see menu items](#)

- Select your choice
- [India](#)
- [Srilanka](#)
- [Australia](#)
- [England](#)

## Footer

## Popup Form

---

### Description

You can display the form in the popup container.

### Example

Following example demonstrates the use of popup form in the jQuery Mobile Framework.

```

<!DOCTYPE html>
<head>
<title>Popup Form</title>
<meta name="viewport" content="width=device-width, initial-scale=1">
<link rel="stylesheet"
href="https://code.jquery.com/mobile/1.4.5/jquery.mobile-1.4.5.min.css">
```

```
<script src="https://code.jquery.com/jquery-1.11.3.min.js"></script>
<script src="https://code.jquery.com/mobile/1.4.5/jquery.mobile-
1.4.5.min.js"></script>
</head>
<body>
<div data-role="page">
    <div data-role="header">
        <h2>Header</h2>
    </div>
    <div id="page1" data-role="main" class="ui-content">
        <a href="#popup_login" data-rel="popup" data-position-to="window"
        class="ui-btn ui-btn-inline ui-icon-home ui-btn-icon-left ui-btn-a" data-
        transition="pop">Login Here</a>
        <div data-role="popup" id="popup_login" data-theme="a">
            <form>
                <div style="padding: 20px 20px;">
                    <h2>Sign in</h2>
                    <label for="name" class="ui-hidden-accessible">Username:</label>
                    <input type="text" name="user" id="name" value=""
                    placeholder="enter the username" data-theme="a">
                    <label for="pwd" class="ui-hidden-accessible">Password:</label>
                    <input type="password" name="pass" id="pwd" value=""
                    placeholder="enter the password" data-theme="a">
                    <button type="submit" class="ui-btn ui-btn-b ui-btn-icon-left ui-
                    icon-forward">Sign in</button>
                </div>
            </form>
        </div>
    </div>
    <div data-role="footer">
        <h2>Footer</h2>
    </div>
</div>
</body>
</html>
```

## Output

Let's carry out the following steps to see how the above code works:

- Save the above html code as **jqm\_popup\_form.html** file in your server root folder.
- Open this HTML file as [http://localhost/jqm\\_popup\\_form.html](http://localhost/jqm_popup_form.html) and the following output will be displayed.

## Header

[Login Here](#)

## Sign in

Username:  Password:

## Footer

## Popup Dialog

---

### Description

You can place the dialog box inside the popup container by setting **href** to the **id** of the panel.

### Example

Following example demonstrates the use of popup dialog in the jQuery Mobile Framework.

```
<!DOCTYPE html>
<head>
<title>Popup Dialog</title>
<meta name="viewport" content="width=device-width, initial-scale=1">
<link rel="stylesheet"
href="https://code.jquery.com/mobile/1.4.5/jquery.mobile-1.4.5.min.css">
<script src="https://code.jquery.com/jquery-1.11.3.min.js"></script>
<script src="https://code.jquery.com/mobile/1.4.5/jquery.mobile-
1.4.5.min.js"></script>
</head>
<body>
```

```

<div data-role="page">
    <div data-role="header">
        <h2>Header</h2>
    </div>
    <div id="page1" data-role="main" class="ui-content">
        <a href="#popup_dialog" data-rel="popup" data-position-to="window" data-transition="pop" class="ui-btn ui-btn-inline ui-icon-delete ui-btn-icon-left ui-btn-b">Delete</a>

        <div data-role="popup" id="popup_dialog" data-theme="b" data-dismissible="false">
            <div role="main" class="ui-content">
                <h2 class="ui-title">Do you want to delete this page?</h2>
                <a href="#" class="ui-btn ui-btn-inline ui-btn-b" data-rel="back">Cancel</a>
                <a href="#" class="ui-btn ui-btn-inline ui-btn-b" data-rel="back">Delete</a>
            </div>
        </div>
    </div>
    <div data-role="footer">
        <h2>Footer</h2>
    </div>
</div>
</body>
</html>

```

## Output

Let's carry out the following steps to see how the above code works:

- Save the above html code as **jqm\_popup\_dialog.html** file in your server root folder.
- Open this HTML file as [http://localhost/jqm\\_popup\\_dialog.html](http://localhost/jqm_popup_dialog.html) and the following output will be displayed.

## Header

[Delete](#)

**Do you want to delete this page?**

[Cancel](#) [Delete](#)

## Footer

## Popup Padding

---

### Description

You can specify the padding for the text in a popup using the ***ui-content*** class.

### Example

Following example demonstrates the use of popup padding in the jQuery Mobile Framework.

```
<!DOCTYPE html>
<head>
<title>Popup Padding</title>
<meta name="viewport" content="width=device-width, initial-scale=1">
<link rel="stylesheet"
href="https://code.jquery.com/mobile/1.4.5/jquery.mobile-1.4.5.min.css">
<script src="https://code.jquery.com/jquery-1.11.3.min.js"></script>
<script src="https://code.jquery.com/mobile/1.4.5/jquery.mobile-
1.4.5.min.js"></script>
</head>
<body>
<div data-role="page">
<div data-role="header">
<h2>Header</h2>
</div>
<div id="page1" data-role="main" class="ui-content">
<a href="#popup_padding" data-rel="popup" class="ui-btn ui-btn-
inline">Popup Padding</a>
<div data-role="popup" id="popup_padding" class="ui-content">
```

```

    <p>It is a formattted text with padding by using the "ui-content"
class.</p>
</div>
</div>
<div data-role="footer">
    <h2>Footer</h2>
</div>
</div>
</body>
</html>

```

## Output

Let's carry out the following steps to see how the above code works:

- Save the above html code as **jqm\_popup\_padding.html** file in your server root folder.
- Open this HTML file as [http://localhost/jqm\\_popup\\_padding.html](http://localhost/jqm_popup_padding.html) and the following output will be displayed.

## Header

### [Popup Padding](#)

It is a formattted text with padding by using the "ui-content" class.

## Footer

## Popup Position

---

### Description

You can specify the position for popup box using the following three ways:

- **data-position-to="window"**: Popup will be positioned to the window.
- **data-position-to="origin"**: Popup will be positioned over the origin.
- **data-position-to="#specified\_position\_id"**: Popup is positioned over the element with a specified #id.

## Example

Following example demonstrates the use of popup position in the jQuery Mobile Framework.

```
<!DOCTYPE html>
<head>
<title>Popup Position</title>
<meta name="viewport" content="width=device-width, initial-scale=1">
<link rel="stylesheet"
href="https://code.jquery.com/mobile/1.4.5/jquery.mobile-1.4.5.min.css">
<script src="https://code.jquery.com/jquery-1.11.3.min.js"></script>
<script src="https://code.jquery.com/mobile/1.4.5/jquery.mobile-
1.4.5.min.js"></script>
</head>
<body>
<div data-role="page">
<div data-role="header">
    <h2>Header</h2>
</div>
<div id="page1" data-role="main" class="ui-content">
    <a href="#position_window" class="ui-btn ui-btn-inline" data-rel="popup"
data-position-to="window">Window Position</a>
    <a href="#position_origin" class="ui-btn ui-btn-inline" data-rel="popup"
data-position-to="origin">Origin Position</a>
    <a href="#position_selector" class="ui-btn ui-btn-inline" data-
rel="popup" data-position-to="#myheader">Position to Specified Id</a>
    <div data-role="popup" id="position_window" class="ui-content" data-theme="a">
        <p>This is positioned to the window.</p>
    </div>
    <div data-role="popup" id="position_origin" class="ui-content" data-theme="a">
        <p>This is positioned to the origin.</p>
    </div>
    <div data-role="popup" id="position_selector" class="ui-content" data-theme="a">
        <p>This is positioned to specified Id. Popup is positioned over the
element with a specified #id.</p>
    </div>
</div>
<div data-role="footer">
    <h2>Footer</h2>

```

```

    </div>
</div>
</body>
</html>
```

## Output

Let's carry out the following steps to see how above code works:

- Save the above html code as **jqm\_popup\_position.html** file in your server root folder.
- Open this HTML file as [http://localhost/jqm\\_popup\\_position.html](http://localhost/jqm_popup_position.html) and the following output will be displayed.

## Header

### [Window Position](#) [Origin Position](#) [Position to Specified Id](#)

This is positioned to the window.

This is positioned to the origin.

This is positioned to specified Id. Popup is positioned over the element with a specified #id.

## Footer

## Popup Transitions

---

### Description

You can use a variety of effects, which specifies the transition from one page to another page using **data-transition** attribute.

### Example

Following example demonstrates the use of popup transitions in the jQuery Mobile Framework.

```

<!DOCTYPE html>
<head>
<title>Popup Transition</title>
<meta name="viewport" content="width=device-width, initial-scale=1">
```

```

<link rel="stylesheet"
href="https://code.jquery.com/mobile/1.4.5/jquery.mobile-1.4.5.min.css">
<script src="https://code.jquery.com/jquery-1.11.3.min.js"></script>
<script src="https://code.jquery.com/mobile/1.4.5/jquery.mobile-
1.4.5.min.js"></script>
</head>
<body>
<div data-role="page">
    <div data-role="header">
        <h2>Header</h2>
    </div>
    <div id="page1" data-role="main" class="ui-content">
        <a href="#my_transition" data-transition="pop" class="ui-btn ui-btn-
        inline" data-rel="popup">Pop</a>
        <a href="#my_transition" data-transition="fade" class="ui-btn ui-btn-
        inline" data-rel="popup">Fade</a>
        <a href="#my_transition" data-transition="flip" class="ui-btn ui-btn-
        inline" data-rel="popup">Flip</a>
        <a href="#my_transition" data-transition="turn" class="ui-btn ui-btn-
        inline" data-rel="popup">Turn</a>
        <a href="#my_transition" data-transition="flow" class="ui-btn ui-btn-
        inline" data-rel="popup">Flow</a>
        <a href="#my_transition" data-transition="slide" class="ui-btn ui-btn-
        inline" data-rel="popup">Slide</a>
        <a href="#my_transition" data-transition="slidedefade" class="ui-btn ui-
        btn-inline" data-rel="popup">Slidedefade</a>
        <a href="#my_transition" data-transition="slideup" class="ui-btn ui-btn-
        inline" data-rel="popup">Slide up</a>
        <a href="#my_transition" data-transition="slidedown" class="ui-btn ui-
        btn-inline" data-rel="popup">Slide down</a>
        <a href="#my_transition" data-transition="none" class="ui-btn ui-btn-
        inline" data-rel="popup">No Transition</a>
        <div data-role="popup" id="my_transition" class="ui-content" data-theme="a">
            <p>Welcome to TutorialsPoint....</p>
        </div>
    </div>
    <div data-role="footer">
        <h2>Footer</h2>
    </div>
</div>
</body>

```

```
</html>
```

## Output

Let's carry out the following steps to see how the above code works:

- Save the above html code as **jqm\_popup\_transitions.html** file in your server root folder.
- Open this HTML file as [http://localhost/jqm\\_popup\\_transitions.html](http://localhost/jqm_popup_transitions.html) and the following output will be displayed.

## Header

[Pop](#) [Fade](#) [Flip](#) [Turn](#) [Flow](#) [Slide](#) [Slidefade](#) [Slide up](#) [Slide down](#) [No Transition](#)

Welcome to Tutorialspoint....

## Footer

## Popup Theme

### Description

You can apply different types of theme for the popup using the following options:

- **data-theme**: It defines the theme for the popup.
- **data-overlay-theme**: It displays the transparent layer behind the popup.

### Example

Following example demonstrates the use of popup theme in the jQuery Mobile Framework.

```
<!DOCTYPE html>
<head>
<title>Popup Theme</title>
<meta name="viewport" content="width=device-width, initial-scale=1">
<link rel="stylesheet"
href="https://code.jquery.com/mobile/1.4.5/jquery.mobile-1.4.5.min.css">
<script src="https://code.jquery.com/jquery-1.11.3.min.js"></script>
<script src="https://code.jquery.com/mobile/1.4.5/jquery.mobile-
1.4.5.min.js"></script>
</head>
```

```

<body>
<div data-role="page">
    <div data-role="header">
        <h2>Header</h2>
    </div>
    <div id="page1" data-role="main" class="ui-content">
        <a href="#themeA_ovrlyA" data-rel="popup" class="ui-btn ui-btn-inline">Theme A and Overlay A</a>
        <div id="themeA_ovrlyA" data-role="popup" data-theme="a" data-overlay-theme="a" class="ui-content">
            <p>This has both theme A and overlay theme A.</p>
        </div>
        <a href="#themeB_ovrlyB" data-rel="popup" class="ui-btn ui-btn-inline">Theme B and overlay A</a>
        <div id="themeB_ovrlyB" data-role="popup" data-overlay-theme="b" data-theme="b" class="ui-content">
            <p>This has both theme B and overlay theme B.</p>
        </div>
    </div>
    <div data-role="footer">
        <h2>Footer</h2>
    </div>
</div>
</body>
</html>

```

## Output

Let's carry out the following steps to see how the above code works:

- Save the above html code as **jqm\_popup\_theme.html** file in your server root folder.
- Open this HTML file as [http://localhost/jqm\\_popup\\_theme.html](http://localhost/jqm_popup_theme.html) and the following output will be displayed.

## Header

### Theme A and Overlay A

This has both theme A and overlay theme A.

### Theme B and overlay A

This has both theme B and overlay theme B.

## Footer

## Popup Arrow

---

### Description

Displays the popup with an arrow and specifies the arrow to the popup using "l(left)", "r(right)", "t(top)" and "b(bottom)".

### Example

Following example demonstrates the use of popup arrow in the jQuery Mobile Framework.

```
<!DOCTYPE html>
<head>
<title>Popup Arrow</title>
<meta name="viewport" content="width=device-width, initial-scale=1">
<link rel="stylesheet"
href="https://code.jquery.com/mobile/1.4.5/jquery.mobile-1.4.5.min.css">
<script src="https://code.jquery.com/jquery-1.11.3.min.js"></script>
<script src="https://code.jquery.com/mobile/1.4.5/jquery.mobile-
1.4.5.min.js"></script>
</head>
<body>
<div data-role="page">
<div data-role="header">
    <h2>Header</h2>
</div>
<div data-role="main" class="ui-content">
    <p>Click the buttons to open a popup with an arrow.</p>

```

```

<a href="#pop1" data-rel="popup" class="ui-btn ui-btn-inline" data-position-to="#arrow_demo">Left</a>
<a href="#pop2" data-rel="popup" class="ui-btn ui-btn-inline" data-position-to="#arrow_demo">Right</a>
<div data-role="popup" id="pop1" class="ui-content" data-arrow="l">
    <p>Arrow on left side..</p>
</div>
<div data-role="popup" id="pop2" class="ui-content" data-arrow="r">
    <p>Arrow on right side.</p>
</div>
<p>Welcome to <span id="arrow_demo" style="color:red;">Tutorialspoint...</span>.</p>
</div>
<div data-role="footer">
    <h2>Footer</h2>
</div>
</div>
</body>
</html>

```

## Output

Let's carry out the following steps to see how the above code works:

- Save the above html code as **jqm\_popup\_arrow.html** file in your server root folder.
- Open this HTML file as [http://localhost/jqm\\_popup\\_arrow.html](http://localhost/jqm_popup_arrow.html) and the following output will be displayed.

## Header

Click the buttons to open a popup with an arrow.

[Left](#) [Right](#)

Arrow on left side..

Arrow on right side..

Welcome to **Tutorialspoint....**

## Footer

## Pre-rendered Popup Markup

---

### Description

You can use pre-rendered markup in the popup to save the startup time.

### Example

Following example demonstrates the use of popup pre-rendered markup in the jQuery Mobile Framework.

```
<!DOCTYPE html>
<head>
<title>Popup Pre-rendered Markup</title>
<meta name="viewport" content="width=device-width, initial-scale=1">
<link rel="stylesheet"
href="https://code.jquery.com/mobile/1.4.5/jquery.mobile-1.4.5.min.css">
<script src="https://code.jquery.com/jquery-1.11.3.min.js"></script>
<script src="https://code.jquery.com/mobile/1.4.5/jquery.mobile-
1.4.5.min.js"></script>
</head>
<body>
<div data-role="page">
<div data-role="header">
<h2>Pre-rendered Popup Markup</h2>
</div>
<div role="main" class="ui-content">
<a href="#pre-rendered" data-rel="popup" class="ui-btn ui-btn-
inline">Open Popup</a>
</div>
<div id="pre-rendered-screen" class="ui-popup-screen ui-screen-
hidden"></div>
<div id="pre-rendered-popup" class="ui-popup-container fade ui-popup-hidden
ui-body-inherit ui-overlay-shadow">
<div id="pre-rendered" class="ui-popup" data-role="popup" data-
transition="slideup">
<p>Welcome to TutorialsPoint...</p>
</div>
</div>
<div data-role="footer">
<h2>Footer</h2>
</div>

```

```
</div>
</body>
</html>
```

## Output

Let's carry out the following steps to see how the above code works:

- Save the above html code as **jqm\_pre-rendered\_popup\_markup.html** file in your server root folder.
- Open this HTML file as [http://localhost/jqm\\_pre-rendered\\_popup\\_markup.html](http://localhost/jqm_pre-rendered_popup_markup.html) and the following output will be displayed.

## Pre-rendered Popup Markup

[Open Popup](#)

Welcome to TutorialsPoint...

## Footer

## Popup Alignment

---

### Description

Sets the alignment along with x-axis or y-axis for the popup in a page.

### Example

Following example demonstrates the use of popup alignment in the jQuery Mobile Framework.

```
<!DOCTYPE html>
<head>
<title>Popup Alignment</title>
<meta name="viewport" content="width=device-width, initial-scale=1">
<link rel="stylesheet"
href="https://code.jquery.com/mobile/1.4.5/jquery.mobile-1.4.5.min.css">
<script src="https://code.jquery.com/jquery-1.11.3.min.js"></script>
<script src="https://code.jquery.com/mobile/1.4.5/jquery.mobile-
1.4.5.min.js"></script>
```

```

<script src="http://demos.jquerymobile.com/1.4.5/popup-
alignment/popup.alignment.js"></script>

<script>
( function( $, undefined ) {
    $.mobile.document.on( "slidestop", function() {
        setTimeout( function() {
            $( "#alignment_popup" ).popup( "option", "align",
                $( "#x-axis" ).val() + "," + $( "#y-axis" ).val() );
        }, 300 );
    });
})( jQuery );
</script>
<style>
#alignment_popup {
    min-width: 200px;
    opacity: 0.8;
}
#alignment_popup1 {
    position: relative;
    left: 50%;
}
</style>
</head>
<body>
<div role="main" class="ui-content">
    <div data-demo-html="true" data-demo-js="#extension">
        <div data-role="popup" id="alignment_popup" class="ui-content">
            <form data-role="fieldset">
                <div>
                    <label for="xalign">X Alignment</label>
                    <input type="range" id="x-axis" value="0.5" min="1" max="3"
step="0.5">
                </div>
                <div>
                    <label for="yalign">Y Alignment</label>
                    <input type="range" id="y-axis" value="0.5" min="1" max="3"
step="0.5">
                </div>
            </form>
        </div>
    </div>
</div>

```

```

</form>
</div>
</div>
<a href="#alignment_popup" id="alignment_popup1" data-rel="popup"
role="button" class="ui-btn ui-btn-inline">Click to Open Popup</a>
</div>
</body>
</html>

```

## Output

Let's carry out the following steps to see how the above code works:

- Save the above html code as **jqm\_popup\_alignment.html** file in your server root folder.
- Open this HTML file as [http://localhost/jqm\\_popup\\_alignment.html](http://localhost/jqm_popup_alignment.html) and the following output will be displayed.



## Dynamic Popup

### Description

Creates the popup dynamically in which the popup gets the right size and position.

### Example

Following example demonstrates the use of dynamic popup in the jQuery Mobile Framework.

```

<!DOCTYPE html>
<head>
<title>Dynamic Popup</title>

```

```

<meta name="viewport" content="width=device-width, initial-scale=1">
<link rel="stylesheet"
href="https://code.jquery.com/mobile/1.4.5/jquery.mobile-1.4.5.min.css">
<script src="https://code.jquery.com/jquery-1.11.3.min.js"></script>
<script src="https://code.jquery.com/mobile/1.4.5/jquery.mobile-
1.4.5.min.js"></script>
<script>
$( document ).on( "pagecreate", "#demo-page", function() {
  $( ".nature_view" ).on( "click", function() {
    var target = $( this ),
      img1 = target.find( "h2" ).html(),
      img2 = target.find( "p" ).html(),
      img3 = target.attr( "id" ),
      closebtn = '<a href="#" data-rel="back" class="ui-btn ui-btn-a ui-icon-
delete ui-btn-icon-notext ui-btn-right">Close</a>',
      header = '<div data-role="header"><h2>' + img1 + ' ' + img2 +
'</h2></div>',
      img = '',
      popup = '<div data-role="popup" id="popup-' + img3 + '" data-short=""'
+ img3 +'" data-theme="none" data-overlay-theme="a"></div>';
    $( header )
      .appendTo( $( popup ) )
      .appendTo( $.mobile.activePage )
      .popup()
      .toolbar()
      .before( closebtn )
      .after( img );
    $( ".img_view", "#popup-" + img3 ).load(function() {
      $( "#popup-" + img3 ).popup( "open" );
      clearTimeout( fallback );
    });
    var fallback = setTimeout(function() {
      $( "#popup-" + img3 ).popup( "open" );
    }, 2000);
  });
  $( document ).on( "popupbeforeposition", ".ui-popup", function() {
    var image = $( this ).children( "img" ),
      height = image.height(),

```

```

width = image.width();
$( this ).attr({ "height": height, "width": width });
var maxHeight = $( window ).height() - 75 + "px";
$( "img.img_view", this ).css( "max-height", maxHeight );
});
$( document ).on( "popupafterclose", ".ui-popup", function() {
$( this ).remove();
});
});
</script>
</head>
<body>
<div data-role="page" id="demo-page" data-url="demo-page">
<div data-role="header" data-theme="b">
<h2>Header</h2>
</div>
<div role="main" class="ui-content">
<ul data-role="listview">
<li><a href="#" class="nature_view"><h2>Wonderful</h2><p>Nature</p></a></li>
</ul>
</div>
<div data-role="header" data-theme="b">
<h2>Footer</h2>
</div>
</div>
</body>
</html>

```

## Output

Let's carry out the following steps to see how the above code works:

- Save the above html code as **jqm\_dynamic\_popup.html** file in your server root folder.

- Open this HTML file as [http://localhost/jqm\\_dynamic\\_popup.html](http://localhost/jqm_dynamic_popup.html) and the following output will be displayed.

## Header



## Wonderful

Nature

## Footer

## iframes in Popup

### Description

It is used to include an iframe inside a popup such as videos and maps.

Following table lists some of the iframe types, which are supported by the jQuery Mobile.

| Sr. No. | Type & Description |
|---------|--------------------|
|---------|--------------------|

|   |  |
|---|--|
| 1 | <b><u>iframes Video</u></b><br>Specifies an inline frame that embeds a document inside another document. |
| 2 | <b><u>Map Popup</u></b><br>You can use the map to specify the Google Maps API.                           |

## iframe Video

---

### Description

You can specify an inline frame that embeds a document inside another document, such as the content of another source, an advertisement, etc.

### Example

Following example demonstrates the use of iframe video in the jQuery Mobile Framework.

```
<!DOCTYPE html>
<head>
<title>iframe Video</title>
<meta name="viewport" content="width=device-width, initial-scale=1">
<link rel="stylesheet"
href="https://code.jquery.com/mobile/1.4.5/jquery.mobile-1.4.5.min.css">
<script src="https://code.jquery.com/jquery-1.11.3.min.js"></script>
<script src="https://code.jquery.com/mobile/1.4.5/jquery.mobile-
1.4.5.min.js"></script>
<script>
$( document ).on( "pagecreate", function() {
    function scale( width, height, padding, border ) {
        var scrWidth = $( window ).width() - 30,
            scrHeight = $( window ).height() - 30,
            ifrPadding = 2 * padding,
            ifrBorder = 2 * border,
            ifrWidth = width + ifrPadding + ifrBorder,
            ifrHeight = height + ifrPadding + ifrBorder,
            h, w;
        if ( ifrWidth < scrWidth && ifrHeight < scrHeight ) {
            w = ifrWidth;
            h = ifrHeight;
        } else if ( ( ifrWidth / scrWidth ) > ( ifrHeight / scrHeight ) ) {
            w = scrWidth;
        }
    }
});</script>
```

```

        h = ( scrWidth / ifrWidth ) * ifrHeight;
    } else {
        h = scrHeight;
        w = ( scrHeight / ifrHeight ) * ifrWidth;
    }
    return {
        'width': w - ( ifrPadding + ifrBorder ),
        'height': h - ( ifrPadding + ifrBorder )
    };
};

$( ".ui-popup iframe" )
.attr( "width", 0 )
.attr( "height", "auto" );
$( "#popup_video" ).on({
    popupbeforeposition: function() {
        // here calling custom function scale() to get the width and height
        var size = scale( 497, 298, 15, 1 ),
            w = size.width,
            h = size.height;
        $( "#popup_video iframe" )
            .attr( "width", w )
            .attr( "height", h );
    },
    popupafterclose: function() {
        $( "#popup_video iframe" )
            .attr( "width", 0 )
            .attr( "height", 0 );
    }
});
});

</script>
</head>
<body>
<div data-role="page">
    <div data-role="header">
        <h2>Header</h2>
    </div>

```

```

<a href="#popup_video" data-rel="popup" data-position-to="window" class="ui-btn ui-btn-inline">Play Video</a>

<div data-role="popup" id="popup_video" data-theme="a" data-tolerance="15,15" class="ui-content">
    <iframe src="/jquery_mobile/images/video.mp4" width="400" height="200"></iframe>
</div>
<div data-role="footer">
    <h2>Footer</h2>
</div>
</div>
</body>
</html>

```

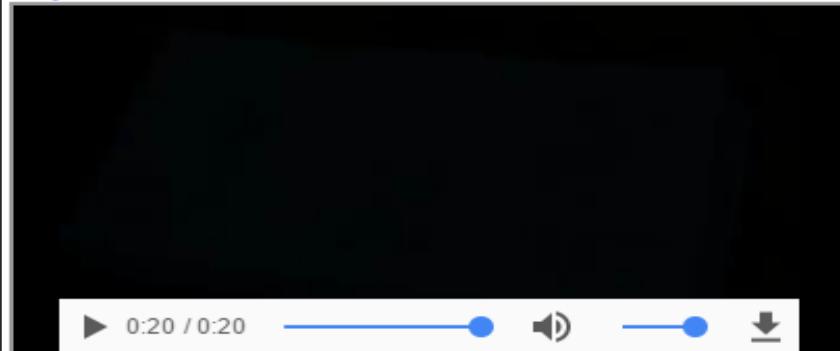
## Output

Let's carry out the following steps to see how the above code works:

- Save the above html code as **jqm\_iframes\_video.html** file in your server root folder.
- Open this HTML file as [http://localhost/jqm\\_iframes\\_video.html](http://localhost/jqm_iframes_video.html) and the following output will be displayed.

## Header

[Play Video](#)



## Footer

## Map Popup

### Description

You can use **iframe** element to specify the Google Maps API.

## Example

Following example demonstrates the use of map popup in the jQuery Mobile Framework.

```
<!DOCTYPE html>
<head>
    <title>Map Popup</title>
    <meta name="viewport" content="width=device-width, initial-scale=1">
    <link rel="stylesheet"
    href="https://code.jquery.com/mobile/1.4.5/jquery.mobile-1.4.5.min.css">
    <script src="https://code.jquery.com/jquery-1.11.3.min.js"></script>
    <script src="https://code.jquery.com/mobile/1.4.5/jquery.mobile-
1.4.5.min.js"></script>
<script>
$( document ).on( "pagecreate", function() {
    function scale( width, height, padding, border ) {
        var scrWidth = $( window ).width() - 30,
            scrHeight = $( window ).height() - 30,
            ifrPadding = 2 * padding,
            ifrBorder = 2 * border,
            ifrWidth = width + ifrPadding + ifrBorder,
            ifrHeight = height + ifrPadding + ifrBorder,
            h, w;
        if ( ifrWidth < scrWidth && ifrHeight < scrHeight ) {
            w = ifrWidth;
            h = ifrHeight;
        } else if ( ( ifrWidth / scrWidth ) > ( ifrHeight / scrHeight ) ) {
            w = scrWidth;
            h = ( scrWidth / ifrWidth ) * ifrHeight;
        } else {
            h = scrHeight;
            w = ( scrHeight / ifrHeight ) * ifrWidth;
        }
        return {
            'width': w - ( ifrPadding + ifrBorder ),
            'height': h - ( ifrPadding + ifrBorder )
        };
    }
    $( ".ui-popup iframe" )
});
```

```

.attr( "width", 0 )
.attr( "height", "auto" );
$( "#popup_pap_iframe" ).contents().find( "#map_popup" )
.css( { "width" : 0, "height" : 0 } );
$( "#popup_pap" ).on({
popupbeforeposition: function() {
    var size = scale( 480, 320, 0, 1 ),
    w = size.width,
    h = size.height;
    $( "#popup_pap_iframe" )
        .attr( "width", w )
        .attr( "height", h );
    $( "#popup_pap_iframe" ).contents().find( "#map_popup" )
        .css( { "width": w, "height" : h } );
},
popupafterclose: function() {
    $( "#popup_pap_iframe" )
        .attr( "width", 0 )
        .attr( "height", 0 );
    $( "#popup_pap_iframe" ).contents().find( "#map_popup" )
        .css( { "width": 0, "height" : 0 } );
}
});
});

```

</script>

</head>

<body>

<div data-role="page">

<div data-role="header">

<h2>Header</h2>

</div>

<a href="#popup\_pap" data-rel="popup" data-position-to="window" class="ui-btn ui-btn-inline">Open Map</a>

<div data-role="popup" id="popup\_pap" data-theme="a">

<a href="#" data-rel="back" class="ui-btn ui-btn-b ui-btn-a ui-icon-delete ui-btn-icon-notext ui-btn-right">Close </a>

<iframe src="/jquery\_mobile/src/map.html" width="480" height="320"></iframe>

```

</div>
<div data-role="footer">
    <h2>Footer</h2>
</div>
</div>
</body>
</html>

```

## Output

Let's carry out the following steps to see how the above code works:

- Save the above html code as **jqm\_map\_popup.html** file in your server root folder.
- Open this HTML file as [http://localhost/jqm\\_map\\_popup.html](http://localhost/jqm_map_popup.html) and the following output will be displayed.

**Header**

[Open Map](#)

[Close](#)

**Footer**

## Scaling Images

### Description

You can scale the image in landscape and portrait format.

### Example

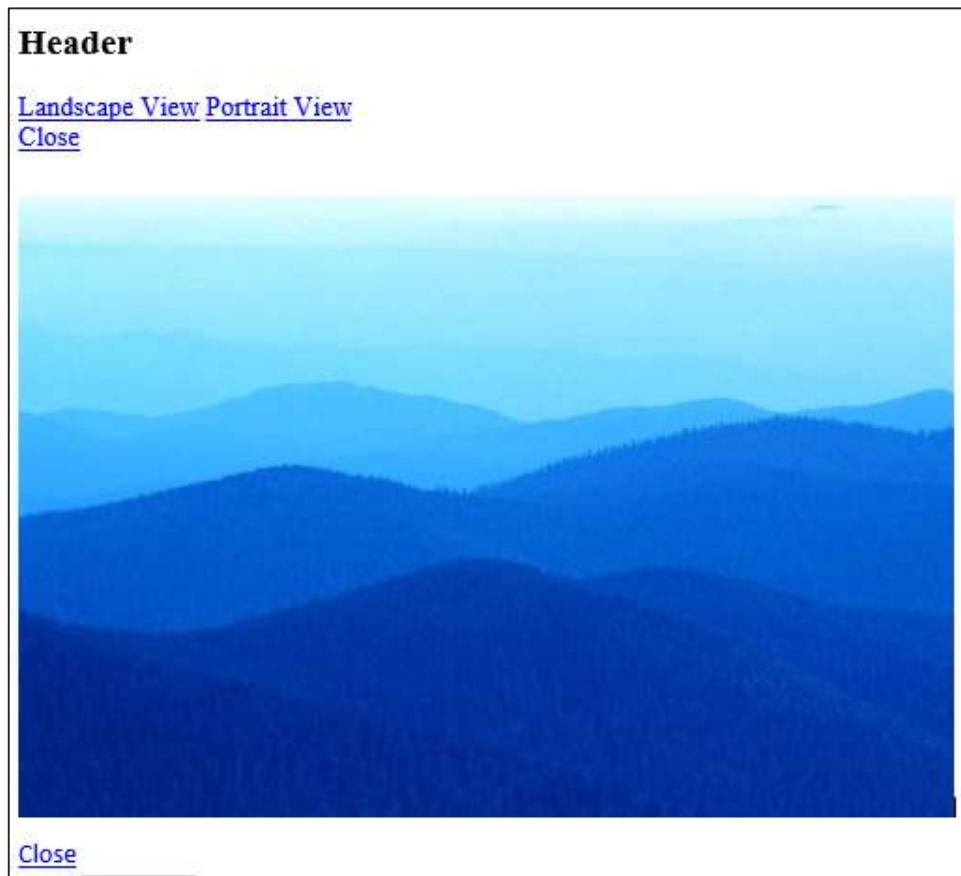
Following example demonstrates the use of scaling images in the jQuery Mobile Framework.

```
<!DOCTYPE html>
<head>
<title>Scaling Images</title>
<meta name="viewport" content="width=device-width, initial-scale=1">
<link rel="stylesheet"
href="https://code.jquery.com/mobile/1.4.5/jquery.mobile-1.4.5.min.css">
<script src="https://code.jquery.com/jquery-1.11.3.min.js"></script>
<script src="https://code.jquery.com/mobile/1.4.5/jquery.mobile-
1.4.5.min.js"></script>
<script src="http://demos.jquerymobile.com/1.4.5/popup-image-scaling/popup-
image-scaling.js"></script>
</head>
<body>
<div data-role="header">
    <h2>Header</h2>
</div>
<a href="#popup_landscape" data-rel="popup" data-position-to="window"
class="ui-btn ui-btn-inline">Landscape View</a>
    <a href="#popup_portrait" data-rel="popup" data-position-to="window"
class="ui-btn ui-btn-inline" data-transition="fade">Portrait View</a>
        <div data-role="popup" id="popup_landscape" class="photopopup" data-overlay-
theme="a" data-tolerance="30,15">
            <a href="#" data-rel="back" class="ui-btn ui-btn-a ui-icon-delete ui-btn-
icon-notext ui-btn-right">Close</a>
        </div>
        <div data-role="popup" id="popup_portrait" class="photopopup" data-overlay-
theme="a" data-tolerance="30,15">
            <a href="#" data-rel="back" class="ui-btn ui-btn-a ui-icon-delete ui-btn-
icon-notext ui-btn-right">Close</a>
        </div>
<div data-role="footer">
    <h2>Footer</h2>
</div>
</body>
</html>
```

## Output

Let's carry out the following steps to see how the above code works:

- Save the above html code as **jqm\_scaling\_images.html** file in your server root folder.
- Open this HTML file as [http://localhost/jqm\\_scaling\\_images.html](http://localhost/jqm_scaling_images.html) and the following output will be displayed.





**Footer**

## Popup on Multiple Pages

### Description

You can use the same popup on the multiple pages by specifying the popup as a child of the body element using the ***data-demo-html*** attribute.

## Example

Following example demonstrates the use of popup across multiple pages in the jQuery Mobile Framework.

```
<!DOCTYPE html>
<head>
<title>Popup on Multiple Pages</title>
<meta name="viewport" content="width=device-width, initial-scale=1">
<link rel="stylesheet"
href="https://code.jquery.com/mobile/1.4.5/jquery.mobile-1.4.5.min.css">
<script src="https://code.jquery.com/jquery-1.11.3.min.js"></script>
<script src="https://code.jquery.com/mobile/1.4.5/jquery.mobile-
1.4.5.min.js"></script>
<script id="popup_script">
$(function(){
    $( "#popup_page" ).enhanceWithin().popup();
});
</script>
</head>
<body>
<div id="popup_page" data-theme="a">
    <ul data-role="listview">
        <li>Global Menu</li>
        <li><a href="#first_page">First Page</a></li>
        <li><a href="#second_page">Second Page</a></li>
        <li><a href="#third_page">Third Page</a></li>
    </ul>
</div>
<div data-role="page" id="first_page" class="jqm-demos">
    <div role="main" class="ui-content jqm-content">
        <h2>Example of same popup on multiple pages</h2>
        <div data-demo-html="#popup_page,#second_page,#third_page" data-demo-
js="#popup_script">
            <a href="#popup_page" data-rel="popup" class="ui-btn ui-btn-inline ui-
corner-all">Click to see menu items</a>
        </div>
    </div>
</div>
<div id="second_page" data-role="page">
```

```
<div data-role="header">
    <a href="#popup_page" data-rel="popup">Menu</a>
    <h2>Second Page</h2>
</div>
<div role="main" class="ui-content">
    <p>This is second page.</p>
</div>
</div>
<div id="third_page" data-role="page">
    <div data-role="header">
        <a href="#popup_page" data-rel="popup">Menu</a>
        <h2>Third Page</h2>
    </div>
    <div role="main" class="ui-content">
        <p>This is third page.</p>
    </div>
</div>
</body>
</html>
```

## Output

Let's carry out the following steps to see how the above code works:

- Save the above html code as **jqm\_popup\_across\_multiple\_pages.html** file in your server root folder.
- Open this HTML file as [http://localhost/jqm\\_popup\\_across\\_multiple\\_pages.html](http://localhost/jqm_popup_across_multiple_pages.html) and the following output will be displayed.

- Global Menu
- [First Page](#)
- [Second Page](#)
- [Third Page](#)

## Example of same popup on multiple pages

[Click to see menu items](#)  
[Menu](#)

### Second Page

This is second page.

[Menu](#)

### Third Page

This is third page.

## Range Slider Widget

---

### Description

Rangeslider widget provides you with a pair of handles allowing you to select a numeric value range.

Following table lists the range slider types used in jQuery mobile.

| Sr. No. | Rangeslider Types & Description  |
|---------|--|
| 1       | <b>Basic rangeslider</b><br>You can use rangeslider widget by adding the <i>data-role</i> attribute and setting it to rangeslider.   |
| 2       | <b>Step attribute</b><br>You can force the slider to snap to a specific increment by adding the <i>step</i> attribute to the input.  |
| 3       | <b>No highlight</b><br>To have <i>no</i> highlight on the track up to the handle position of the slider, add the <i>data-highlight</i> attribute and set to <i>false</i> . |

|   |  |
|---|--|
| 4 | <b><u>Theme</u></b><br>You can set the theme for your rangeslider using <i>data-track-theme</i> and <i>data-theme</i> attribute.                                   |
| 5 | <b><u>Mini</u></b><br>For more compact version of rangeslider, you can use <i>data-mini</i> attribute.   |
| 6 | <b><u>Disabled</u></b><br>You can disable the rangeslider using <i>disabled</i> attribute.   |
| 7 | <b><u>Label hidden</u></b><br>You can hide the label of the rangeslider by adding the <i>ui-hidden-accessible</i> class to your label.                             |
| 8 | <b><u>Fieldcontain</u></b><br>Adding <i>ui-field-contain</i> class to the <i>div</i> wrappers of form elements will ensure that the form is rendered responsively. |
| 9 | <b><u>Fieldcontain, mini</u></b><br>The <i>ui-field-contain</i> class can be added to <i>mini</i> rangeslider in order to render it responsively.                  |

## Basic Rangeslider

---

### Description

You can use rangeslider widget by adding the *data-role* attribute and setting it to rangeslider.

### Example

Following example demonstrates the use of basic rangeslider in jQuery Mobile.

```
<!DOCTYPE html>
<html>
<head>
    <title>Basic Rangeslider</title>
    <meta name="viewport" content="width=device-width, initial-scale=1">
    <link rel="stylesheet"
    href="https://code.jquery.com/mobile/1.4.5/jquery.mobile-1.4.5.min.css">
    <script src="https://code.jquery.com/jquery-1.11.3.min.js"></script>
    <script src="https://code.jquery.com/mobile/1.4.5/jquery.mobile-
    1.4.5.min.js"></script>
```

```

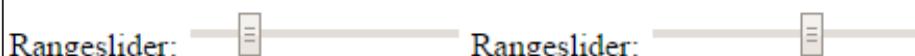
</head>
<body>
<form>
<div data-role="rangeslider">
<label for="range-1">Rangeslider:</label>
<input type="range" name="range-1" id="range-1" min="0" max="100" value="20">
<label for="range-2">Rangeslider:</label>
<input type="range" name="range-2" id="range-2" min="0" max="100" value="60">
</div>
</form>
</body>
</html>

```

## Output

Let's carry out the following steps to see how the above code works:

- Save the above html code as **range\_slider\_basic.html** file in your server root folder.
- Open this HTML file as [http://localhost/range\\_slider\\_basic.html](http://localhost/range_slider_basic.html) and the following output will be displayed.



## Rangeslider Step Attribute

---

### Description

You can force the slider to snap to a specific increment by adding the *step* attribute to the input.

## Example

Following example demonstrates the use of rangeslider step attribute in jQuery Mobile.

```
<!DOCTYPE html>
<head>
    <title>Rangeslider Step Attribute</title>
    <meta name="viewport" content="width=device-width, initial-scale=1">
    <link rel="stylesheet"
    href="https://code.jquery.com/mobile/1.4.5/jquery.mobile-1.4.5.min.css">
    <script src="https://code.jquery.com/jquery-1.11.3.min.js"></script>
    <script src="https://code.jquery.com/mobile/1.4.5/jquery.mobile-
1.4.5.min.js"></script>
</head>
<body>
    <form>
        <div data-role="rangeslider">
            <label for="range-1">Rangeslider:</label>
            <input type="range" name="range-1" id="range-1" min="0" max="10"
step=".1" value="6.2">
            <label for="range-2">Rangeslider:</label>
            <input type="range" name="range-2" id="range-2" min="0" max="10"
step=".1" value="7.8">
        </div>
    </form>
</body>
</html>
```

## Output

Let's carry out the following steps to see how the above code works:

- Save the above html code as **range\_slider\_step.html** file in your server root folder.
- Open this HTML file as [http://localhost/range\\_slider\\_step.html](http://localhost/range_slider_step.html) and the following output will be displayed.



## Rangeslider No Highlight

### Description

To have no highlight on the track up to the handle position of the slider, add the ***data-highlight*** attribute and set to *false*.

### Example

Following example demonstrates the use of rangeslider with no highlight in jQuery Mobile.

```
<!DOCTYPE html>
<head>
    <title>Rangeslider No Highlight</title>
    <meta name="viewport" content="width=device-width, initial-scale=1">
    <link rel="stylesheet"
    href="https://code.jquery.com/mobile/1.4.5/jquery.mobile-1.4.5.min.css">
    <script src="https://code.jquery.com/jquery-1.11.3.min.js"></script>
    <script src="https://code.jquery.com/mobile/1.4.5/jquery.mobile-
1.4.5.min.js"></script>
</head>
<body>
    <form>
        <div data-role="rangeslider" data-highlight="false">
            <label for="range-1">Rangeslider:</label>
            <input type="range" name="range-1" id="range-1" min="0" max="100" value="40">
            <label for="range-2">Rangeslider:</label>
            <input type="range" name="range-2" id="range-2" min="0" max="100" value="70">
        </div>
    </form>
</body>
</html>
```

## Output

Let's carry out the following steps to see how the above code works:

- Save the above html code as **range\_slider\_nohighlight.html** file in your server root folder.
- Open this HTML file as [http://localhost/range\\_slider\\_nohighlight.html](http://localhost/range_slider_nohighlight.html) and the following output will be displayed.



## Rangeslider Theme

---

### Description

You can set the theme for your rangeslider using ***data-track-theme*** and ***data-theme*** attribute.

### Example

Following example demonstrates the use of rangeslider theme in jQuery Mobile.

```
<!DOCTYPE html>
<head>
  <title>Rangeslider Theme</title>
  <meta name="viewport" content="width=device-width, initial-scale=1">
  <link rel="stylesheet"
  href="https://code.jquery.com/mobile/1.4.5/jquery.mobile-1.4.5.min.css">
  <script src="https://code.jquery.com/jquery-1.11.3.min.js"></script>
  <script src="https://code.jquery.com/mobile/1.4.5/jquery.mobile-
1.4.5.min.js"></script>
</head>
<body>
  <form>
    <div data-role="rangeslider" data-track-theme="b" data-theme="a">
```

```

<label for="range-1">Rangeslider:</label>
<input type="range" name="range-1" id="range-1" min="0" max="100" value="30">
<label for="range-2">Rangeslider:</label>
<input type="range" name="range-2" id="range-2" min="0" max="100" value="60">
</div>
</form>
</body>
</html>

```

## Output

Let's carry out the following steps to see how the above code works:

- Save the above html code as **range\_slider\_theme.html** file in your server root folder.
- Open this HTML file as [http://localhost/range\\_slider\\_theme.html](http://localhost/range_slider_theme.html) and the following output will be displayed.

## Rangeslider Mini

---

### Description

For more compact version of rangeslider, you can use **data-mini** attribute. It is very useful when you want to display rangeslider in less space.

### Example

Following example demonstrates the use of mini rangeslider in jQuery Mobile.

```

<!DOCTYPE html>
<head>
    <title>Rangeslider Mini</title>
    <meta name="viewport" content="width=device-width, initial-scale=1">
    <link rel="stylesheet"
    href="https://code.jquery.com/mobile/1.4.5/jquery.mobile-1.4.5.min.css">

```

```

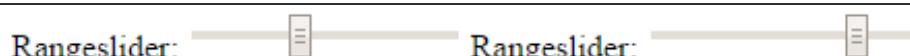
<script src="https://code.jquery.com/jquery-1.11.3.min.js"></script>
<script src="https://code.jquery.com/mobile/1.4.5/jquery.mobile-
1.4.5.min.js"></script>
</head>
<body>
<form>
<div data-role="rangeslider" data-mini="true">
<label for="range-1">Rangeslider:</label>
<input type="range" name="range-1" id="range-1" min="0" max="100" value="40">
<label for="range-2">Rangeslider:</label>
<input type="range" name="range-2" id="range-2" min="0" max="100" value="80">
</div>
</form>
</body>
</html>

```

## Output

Let's carry out the following steps to see how the above code works:

- Save the above html code as **range\_slider\_mini.html** file in your server root folder.
- Open this HTML file as [http://localhost/range\\_slider\\_mini.html](http://localhost/range_slider_mini.html) and the following output will be displayed.



## Rangeslider Disabled

### Description

You can disable the rangeslider using **disabled** attribute.

## Example

Following example demonstrates the use of disabled rangeslider in jQuery Mobile.

```
<!DOCTYPE html>
<head>
    <title>Rangeslider Disabled</title>
    <meta name="viewport" content="width=device-width, initial-scale=1">
    <link rel="stylesheet"
    href="https://code.jquery.com/mobile/1.4.5/jquery.mobile-1.4.5.min.css">
    <script src="https://code.jquery.com/jquery-1.11.3.min.js"></script>
    <script src="https://code.jquery.com/mobile/1.4.5/jquery.mobile-
1.4.5.min.js"></script>
</head>
<body>
    <form>
        <div data-role="rangeslider">
            <label for="range-1">Rangeslider:</label>
            <input type="range" name="range-1" id="range-1" disabled="disabled"
min="0" max="100" value="30">
            <label for="range-2">Rangeslider:</label>
            <input type="range" name="range-2" id="range-2" disabled="disabled"
min="0" max="100" value="60">
        </div>
    </form>
</body>
</html>
```

## Output

Let's carry out the following steps to see how the above code works:

- Save the above html code as **range\_slider\_disabled.html** file in your server root folder.
- Open this HTML file as [http://localhost/range\\_slider\\_disabled.html](http://localhost/range_slider_disabled.html) and the following output will be displayed.

Rangeslider: Rangeslider:

## Rangeslider Label Hidden

### Description

You can hide the label of the rangeslider by adding the ***ui-hidden-accessible*** class to your label.

### Example

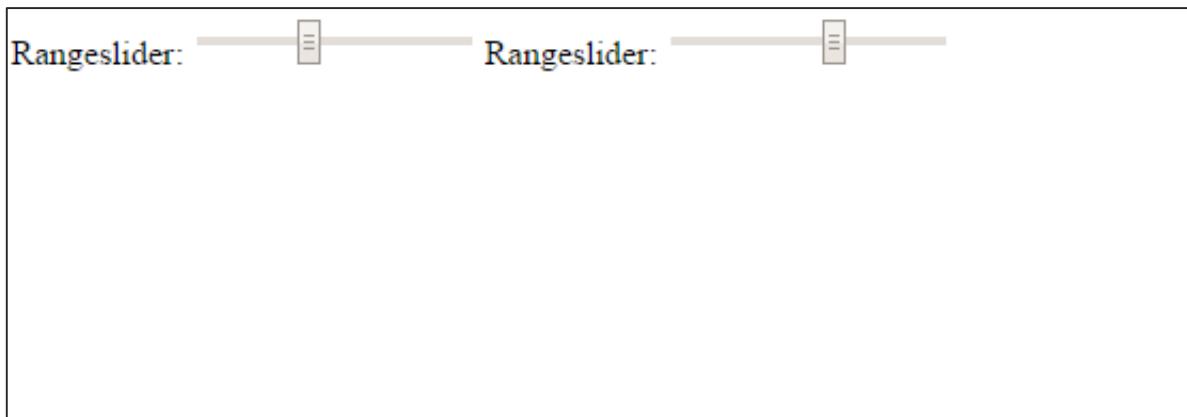
Following example demonstrates the use of rangeslider with hidden label in jQuery Mobile.

```
<!DOCTYPE html>
<head>
    <title>Rangeslider Label Hidden</title>
    <meta name="viewport" content="width=device-width, initial-scale=1">
    <link rel="stylesheet"
    href="https://code.jquery.com/mobile/1.4.5/jquery.mobile-1.4.5.min.css">
    <script src="https://code.jquery.com/jquery-1.11.3.min.js"></script>
    <script src="https://code.jquery.com/mobile/1.4.5/jquery.mobile-
    1.4.5.min.js"></script>
</head>
<body>
    <form>
        <div data-role="rangeslider">
            <label for="range-1" class="ui-hidden-accessible">Rangeslider:</label>
            <input type="range" name="range-1" id="range-1" min="0" max="100" value="40">
            <label for="range-2">Rangeslider:</label>
            <input type="range" name="range-2" id="range-2" min="0" max="100" value="60">
        </div>
    </form>
</body>
</html>
```

## Output

Let's carry out the following steps to see how the above code works:

- Save the above html code as **range\_slider\_label\_hidden.html** file in your server root folder.
- Open this HTML file as [http://localhost/range\\_slider\\_label\\_hidden.html](http://localhost/range_slider_label_hidden.html) and the following output will be displayed.



## Rangeslider Fieldcontain

---

### Description

You can add **ui-field-contain** class to **div** wrappers of individual elements of the form which includes a label. The class and the **div** make sure that the form is responsively rendered.

### Example

Following example demonstrates the use of fieldcontain in jQuery Mobile.

```
<!DOCTYPE html>
<head>
  <title>Rangeslider Fieldcontain</title>
  <meta name="viewport" content="width=device-width, initial-scale=1">
  <link rel="stylesheet"
  href="https://code.jquery.com/mobile/1.4.5/jquery.mobile-1.4.5.min.css">
  <script src="https://code.jquery.com/jquery-1.11.3.min.js"></script>
  <script src="https://code.jquery.com/mobile/1.4.5/jquery.mobile-
  1.4.5.min.js"></script>
</head>
<body>
<form>
```

```

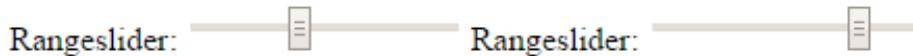
<div class="ui-field-contain">
    <div data-role="rangeslider">
        <label for="range-1">Rangeslider:</label>
        <input type="range" name="range-1" id="range-1" min="0" max="100"
value="40">
        <label for="range-2">Rangeslider:</label>
        <input type="range" name="range-2" id="range-2" min="0" max="100"
value="80">
    </div>
</div>
</form>
</body>
</html>

```

## Output

Let's carry out the following steps to see how the above code works:

- Save the above html code as **range\_slider\_fieldcontain.html** file in your server root folder.
- Open this HTML file as [http://localhost/range\\_slider\\_fieldcontain.html](http://localhost/range_slider_fieldcontain.html) and the following output will be displayed.



## Rangeslider Fieldcontain Mini

---

### Description

The **ui-field-contain** class can be added to **mini** rangeslider in order to render it responsively.

## Example

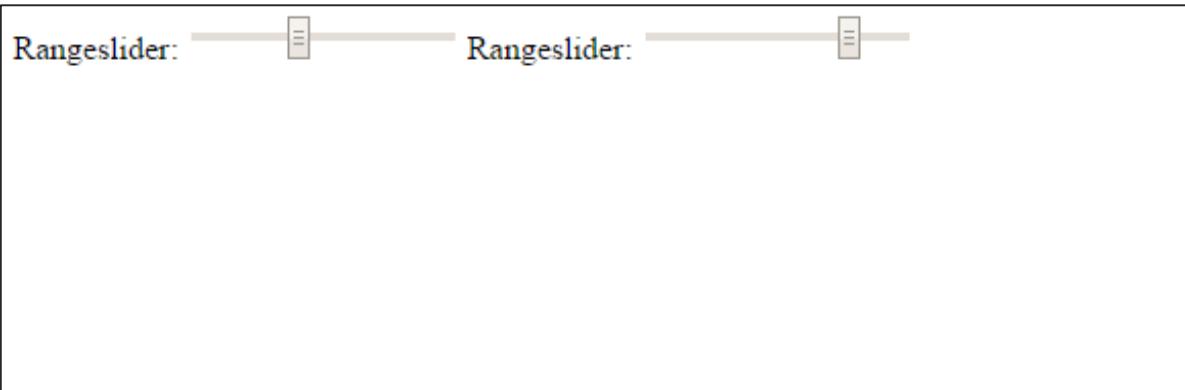
Following example demonstrates the use of fieldcontain class to mini rangesilder in jQuery Mobile.

```
<!DOCTYPE html>
<head>
    <title>Fieldcontain Mini Rangeslider</title>
    <meta name="viewport" content="width=device-width, initial-scale=1">
    <link rel="stylesheet"
    href="https://code.jquery.com/mobile/1.4.5/jquery.mobile-1.4.5.min.css">
    <script src="https://code.jquery.com/jquery-1.11.3.min.js"></script>
    <script src="https://code.jquery.com/mobile/1.4.5/jquery.mobile-
1.4.5.min.js"></script>
</head>
<body>
    <form>
        <div class="ui-field-contain">
            <div data-role="rangeslider" data-mini="true">
                <label for="range-1">Rangeslider:</label>
                <input type="range" name="range-1" id="range-1" min="0" max="100"
value="40">
                <label for="range-2">Rangeslider:</label>
                <input type="range" name="range-2" id="range-2" min="0" max="100"
value="80">
            </div>
        </div>
    </form>
</body>
</html>
```

## Output

Let's carry out the following steps to see how the above code works:

- Save the above html code as **range\_slider\_fieldcontain\_mini.html** file in your server root folder.
- Open this HTML file as [http://localhost/range\\_slider\\_fieldcontain\\_mini.html](http://localhost/range_slider_fieldcontain_mini.html) and the following output will be displayed.



## Table Widget

---

### Description

jQuery Mobile uses a table to represent the data in terms of rows and columns, i.e. displays the data in tabular format.

Following table lists some of the table types, which are supported by the jQuery Mobile.

| Sr. No. | Type & Description   |
|---------|--|
| 1       | <b><u>Table Column Toggle</u></b><br>The column toggle puts the columns in a hiding place and allows the user to select columns as per their choice.                     |
| 2       | <b><u>Column Toggle Heading Groups</u></b><br>Displays the content according to the heading group name.  |
| 3       | <b><u>Column Toggle Customization Options</u></b><br>Specifies the custom options for column toggle table.   |
| 4       | <b><u>Table Reflow</u></b><br>Represents the data in a horizontal format by collapsing the table into stacked representation and groups the rows into a vertical format. |
| 5       | <b><u>Table Reflow Heading Groups</u></b><br>Displays the column headings as sub headings between the data sets and represents the data in terms of value pairs.         |
| 6       | <b><u>Table Reflow Stripes and Strokes</u></b><br>Displays the data sets in stripes and strokes format.  |
| 7       | <b><u>Table Reflow Custom Styles</u></b><br>Specifies the custom styles for the reflow table.  |

## Column Toggle

---

### Description

The column toggle puts the columns in a hiding place and allows the user to select columns as per their choice using the ***data-mode="columntoggle"*** attribute.

### Example

Following example demonstrates the use of column toggle in the jQuery Mobile Framework.

```
<!DOCTYPE html>
<head>
<title>Table Column Toggle</title>
<meta name="viewport" content="width=device-width, initial-scale=1">
<link rel="stylesheet"
href="https://code.jquery.com/mobile/1.4.5/jquery.mobile-1.4.5.min.css">
<script src="https://code.jquery.com/jquery-1.11.3.min.js"></script>
<script src="https://code.jquery.com/mobile/1.4.5/jquery.mobile-
1.4.5.min.js"></script>
</head>
<body>
<div data-role="header">
<h2>Header</h2>
</div>
<table data-role="table" id="table-column-toggle" data-mode="columntoggle"
class="ui-responsive table-stripe" data-column-btn-theme="b" data-column-btn-
text="Click here to display columns...">
<thead>
<tr>
<th data-priority="1">Position</th>
<th>Team</th>
<th data-priority="2">Rating</th>
<th data-priority="3">Points</th>
</tr>
</thead>
<tbody>
<tr>
<th>1</th>
<td>SOUTH AFRICA</td>
```

```
<td>114</td>
<td>3308</td>
</tr>
<tr>
<th>2</th>
<td>INDIA</td>
<td>110</td>
<td>3535</td>
</tr>
<tr>
<th>3</th>
<td>AUSTRALIA</td>
<td>109</td>
<td>4376</td>
</tr>
<tr>
<th>4</th>
<td>PAKISTAN</td>
<td>106</td>
<td>2977</td>
</tr>
<tr>
<th>5</th>
<td>NEW ZEALAND</td>
<td>99</td>
<td>3578</td>
</tr>
<tr>
<th>6</th>
<td>ENGLAND</td>
<td>99</td>
<td>3940</td>
</tr>
<tr>
<th>7</th>
<td>SRI LANKA</td>
<td>89</td>
```

```

<td>3123</td>
</tr>
<tr>
<th>8</th>
<td>WEST INDIES</td>
<td>76</td>
<td>2504</td>
</tr>
<tr>
<th>9</th>
<td>BANGLADESH</td>
<td>47</td>
<td>1026</td>
</tr>
<tr>
<th>10</th>
<td>ZIMBABWE</td>
<td>5</td>
<td>53</td>
</tr>
</tbody>
</table>
<div data-role="footer">
<h2>Footer</h2>
</div>
</body>
</html>

```

## Output

Let's carry out the following steps to see how the above code works:

- Save the above html code as **jqm\_table\_column\_toggle.html** file in your server root folder.
- Open this HTML file as [http://localhost/jqm\\_table\\_column\\_toggle.html](http://localhost/jqm_table_column_toggle.html) and the following output will be displayed.

## Header

| Position | Team         | Rating | Points |
|----------|--------------|--------|--------|
| 1        | SOUTH AFRICA | 114    | 3308   |
| 2        | INDIA        | 110    | 3535   |
| 3        | AUSTRALIA    | 109    | 4376   |
| 4        | PAKISTAN     | 106    | 2977   |
| 5        | NEW ZEALAND  | 99     | 3578   |
| 6        | ENGLAND      | 99     | 3940   |
| 7        | SRI LANKA    | 89     | 3123   |
| 8        | WEST INDIES  | 76     | 2504   |
| 9        | BANGLADESH   | 47     | 1026   |
| 10       | ZIMBABWE     | 5      | 53     |

## Footer

### Column Chooser Mode and Setting Column Priority

You can display the table element using **data-role="table"** and **data-mode="columntoggle"** attributes as shown in the following tag.

```
<table data-role="table" id="table-column-toggle" data-mode="columntoggle">
```

You can specify which column should be hidden or shown using the **data-priority** attribute and assign the priority value from 1 to 6. The table header which will be having the **data-priority** attribute, will be available in the column menu.

### Theme and Customization

By default, the text on the button is "Columns..." however you can change the text using the **data-column-btn-text** attribute to the table. You can also set the theme for button or column picker menu using the **data-column-btn-theme** attribute. The rows can be displayed in a striped format by adding **table-stripe** class to the table element.

### Making Tables Responsive

You can make the tables responsive according to screen size and orientation. The media queries specify the responsive behavior to columns using priority and writes the media query using **min-width** attribute. The widths which are set using em units, will respond to font size changes and you can calculate the value of pixel in em units by dividing the target width by 16 pixels.

## Applying a Preset Breakpoint

You can apply the custom breakpoints for each priority levels by adding the **`class="ui-responsive"`** to the table element as shown below:

```
<table data-role="table" class="ui-responsive" id="table-column-toggle" data-mode="columntoggle">
```

You can use the following preset styles for each priority levels:

- **`data-priority="1"`**: It is used to display the column at 320px(20em).
- **`data-priority="2"`**: It is used to display the column at 480px(30em).
- **`data-priority="3"`**: It is used to display the column at 640px(40em).
- **`data-priority="4"`**: It is used to display the column at 800px(50em).
- **`data-priority="5"`**: It is used to display the column at 960px(60em).
- **`data-priority="6"`**: It is used to display the column at 1120px(70em).

## Grouped Column Headers

The column data can be shown under each grouped heading for financial data. You can use the priority attribute to parse the specified row and display the heading groups as options. Any table header given a priority will be available in the column picker menu.

## Column Toggle Heading Groups

---

### Description

You can display the content according to the heading group name using the **`data-mode="columntoggle"`** sttribute.

### Example

Following example demonstrates the use of column toggle heading groups in the jQuery Mobile Framework.

```
<!DOCTYPE html>
<head>
<title>Table Column Toggle Heading Group</title>
<meta name="viewport" content="width=device-width, initial-scale=1">
<link rel="stylesheet"
href="https://code.jquery.com/mobile/1.4.5/jquery.mobile-1.4.5.min.css">
<script src="https://code.jquery.com/jquery-1.11.3.min.js"></script>
<script src="https://code.jquery.com/mobile/1.4.5/jquery.mobile-
1.4.5.min.js"></script>
<style>
```

```
.th-groups th {  
    text-align: center;  
    background-color: rgba(0,0,0,.1);  
    border-right: 1px solid #fff;  
}  
  
/* Show priority 1 at 320px*/  
@media screen and (min-width: 20em) {  
    th.ui-table-priority-1,  
    td.ui-table-priority-1 {  
        display: table-cell;  
    }  
}  
  
/* Show priority 2 at 480px*/  
@media screen and (min-width: 30em) {  
    th.ui-table-priority-2,  
    td.ui-table-priority-2 {  
        display: table-cell;  
    }  
}  
  
/* Show priority 3 at 640px*/  
@media screen and (min-width: 40em) {  
    th.ui-table-priority-3,  
    td.ui-table-priority-3 {  
        display: table-cell;  
    }  
}  
  
/* Show priority 4 at 800px*/  
@media screen and (min-width: 50em) {  
    th.ui-table-priority-4,  
    td.ui-table-priority-4 {  
        display: table-cell;  
    }  
}  
  
/* Show priority 5 at 960px*/  
@media screen and (min-width: 60em) {  
    th.ui-table-priority-5,
```

```
td.ui-table-priority-5 {  
    display: table-cell;  
}  
}  
/* Show priority 6 at 1200px*/  
@media screen and (min-width: 75em) {  
    th.ui-table-priority-6,  
    td.ui-table-priority-6 {  
        display: table-cell;  
    }  
}  
</style>  
</head>  
<body>  
<div data-role="header">  
    <h2>Header</h2>  
</div>  
<table data-role="table" id="table_columns" data-column-btn-theme="b" data-  
column-popup-theme="a" data-mode="columntoggle" class="table-stroke" data-  
column-btn-text="Click here to display columns...">  
    <thead>  
        <tr class="th-groups">  
            <td></td>  
            <th colspan="3" data-priority="5">INDIA</th>  
            <th colspan="3" data-priority="4">AUSTRALIA</th>  
            <th colspan="3" data-priority="2">ENGLAND</th>  
        </tr>  
        <tr>  
            <th>Stadium</th>  
            <th>Played</th>  
            <th>Won</th>  
            <th>Loss</th>  
            <th>Played</th>  
            <th>Won</th>  
            <th>Loss</th>  
            <th>Played</th>  
            <th>Won</th>  
        </tr>  
    </thead>  
    <tbody>  
        <tr>  
            <td>Wankhede</td>  
            <td>10</td>  
            <td>5</td>  
            <td>3</td>  
            <td>10</td>  
            <td>5</td>  
            <td>3</td>  
            <td>10</td>  
            <td>5</td>  
        </tr>  
        <tr>  
            <td>Eden Park</td>  
            <td>10</td>  
            <td>6</td>  
            <td>4</td>  
            <td>10</td>  
            <td>6</td>  
            <td>4</td>  
            <td>10</td>  
            <td>6</td>  
        </tr>  
        <tr>  
            <td>Lord's</td>  
            <td>10</td>  
            <td>7</td>  
            <td>3</td>  
            <td>10</td>  
            <td>7</td>  
            <td>3</td>  
            <td>10</td>  
            <td>7</td>  
        </tr>  
    </tbody>  
</table>
```

```
<th>Loss</th>
</tr>
</thead>
<tbody>
<tr>
    <th>Sydney</th>
    <td>20</td>
    <td>14</td>
    <td>6</td>
    <td>15</td>
    <td>10</td>
    <td>5</td>
    <td>10</td>
    <td>3</td>
    <td>7</td>
</tr>
<tr>
    <th>Eden Gardens</th>
    <td>10</td>
    <td>9</td>
    <td>1</td>
    <td>8</td>
    <td>3</td>
    <td>5</td>
    <td>9</td>
    <td>5</td>
    <td>4</td>
</tr>
<tr>
    <th>Oval</th>
    <td>12</td>
    <td>9</td>
    <td>3</td>
    <td>15</td>
    <td>3</td>
    <td>12</td>
    <td>20</td>
```

```

<td>15</td>
<td>5</td>
</tr>
</tbody>
</table>
<div data-role="footer">
  <h2>Footer</h2>
</div>
</body>
</html>

```

## Output

Let's carry out the following steps to see how the above code works:

- Save the above html code as **jqm\_column\_toggle\_heading\_groups.html** file in your server root folder.
- Open this HTML file as [http://localhost/jqm\\_column\\_toggle\\_heading\\_groups.html](http://localhost/jqm_column_toggle_heading_groups.html) and the following output will be displayed.

## Header

|              | INDIA  |     |      | AUSTRALIA |     |      | ENGLAND |     |      |
|--------------|--------|-----|------|-----------|-----|------|---------|-----|------|
| Stadium      | Played | Won | Loss | Played    | Won | Loss | Played  | Won | Loss |
| Sydney       | 20     | 14  | 6    | 15        | 10  | 5    | 10      | 3   | 7    |
| Eden Gardens | 10     | 9   | 1    | 8         | 3   | 5    | 9       | 5   | 4    |
| Oval         | 12     | 9   | 3    | 15        | 3   | 12   | 20      | 15  | 5    |

## Footer

## Column Toggle Customization

### Description

You can specify the custom options for column toggle table using different classes such as **ui-body-c**, **table-stroke** etc.

### Example

Following example demonstrates the use of column toggle customization in the jQuery Mobile Framework.

```
<!DOCTYPE html>
<head>
<title>Table Column Toggle Customization</title>
<meta name="viewport" content="width=device-width, initial-scale=1">
<link rel="stylesheet"
href="https://code.jquery.com/mobile/1.4.5/jquery.mobile-1.4.5.min.css">
<script src="https://code.jquery.com/jquery-1.11.3.min.js"></script>
<script src="https://code.jquery.com/mobile/1.4.5/jquery.mobile-
1.4.5.min.js"></script>
</head>
<body>
<div data-role="header">
    <h2>Header</h2>
</div>
<table data-role="table" id="table_columns" data-mode="columntoggle" class="ui-
body-c table-stroke ui-responsive" data-column-btn-theme="d" data-column-btn-
text="Click here to display columns..." data-column-popup-theme="b">
    <thead>
        <tr class="ui-bar-a">
            <th data-priority="1">Position</th>
            <th>Team</th>
            <th data-priority="2">Rating</th>
            <th data-priority="3">Points</th>
        </tr>
    </thead>
    <tbody>
        <tr>
            <th>1</th>
            <td>SOUTH AFRICA</td>
            <td>114</td>
            <td>3308</td>
        </tr>
        <tr>
            <th>2</th>
            <td>INDIA</td>
            <td>110</td>
```

```
<td>3535</td>
</tr>
<tr>
    <th>3</th>
    <td>AUSTRALIA</td>
    <td>109</td>
    <td>4376</td>
</tr>
<tr>
    <th>4</th>
    <td>PAKISTAN</td>
    <td>106</td>
    <td>2977</td>
</tr>
<tr>
    <th>5</th>
    <td>NEW ZEALAND</td>
    <td>99</td>
    <td>3578</td>
</tr>
<tr>
    <th>6</th>
    <td>ENGLAND</td>
    <td>99</td>
    <td>3940</td>
</tr>
<tr>
    <th>7</th>
    <td>SRI LANKA</td>
    <td>89</td>
    <td>3123</td>
</tr>
<tr>
    <th>8</th>
    <td>WEST INDIES</td>
    <td>76</td>
    <td>2504</td>
```

```
</tr>
<tr>
    <th>9</th>
    <td>BANGLADESH</td>
    <td>47</td>
    <td>1026</td>
</tr>
<tr>
    <th>10</th>
    <td>ZIMBABWE</td>
    <td>5</td>
    <td>53</td>
</tr>
</tbody>
</table>
<div data-role="footer">
    <h2>Footer</h2>
</div>
</body>
</html>
```

## Output

Let's carry out the following steps to see how the above code works:

- Save the above html code as **jqm\_column\_toggle\_customization.html** file in your server root folder.
- Open this HTML file as [http://localhost/jqm\\_column\\_toggle\\_customization.html](http://localhost/jqm_column_toggle_customization.html) and the following output will be displayed.

## Header

| <b>Position</b> | <b>Team</b>  | <b>Rating</b> | <b>Points</b> |
|-----------------|--------------|---------------|---------------|
| 1               | SOUTH AFRICA | 114           | 3308          |
| 2               | INDIA        | 110           | 3535          |
| 3               | AUSTRALIA    | 109           | 4376          |
| 4               | PAKISTAN     | 106           | 2977          |
| 5               | NEW ZEALAND  | 99            | 3578          |
| 6               | ENGLAND      | 99            | 3940          |
| 7               | SRI LANKA    | 89            | 3123          |
| 8               | WEST INDIES  | 76            | 2504          |
| 9               | BANGLADESH   | 47            | 1026          |
| 10              | ZIMBABWE     | 5             | 53            |

## Footer

## Table Reflow

### Description

You can represent the data in a horizontal format by collapsing the table into a stacked representation. This groups the rows into a vertical format using the **data-mode="reflow"** attribute.

### Example

Following example demonstrates the use of table reflow in the jQuery Mobile Framework.

```
<!DOCTYPE html>
<head>
<title>Table Column Toggle Customization</title>
<meta name="viewport" content="width=device-width, initial-scale=1">
<link rel="stylesheet"
href="https://code.jquery.com/mobile/1.4.5/jquery.mobile-1.4.5.min.css">
<script src="https://code.jquery.com/jquery-1.11.3.min.js"></script>
<script src="https://code.jquery.com/mobile/1.4.5/jquery.mobile-
1.4.5.min.js"></script>
</head>
<body>
<div data-role="header">
```

```
<h2>Header</h2>
</div>

<table data-role="table" id="table_columns" data-mode="reflow" class="ui-
responsive">
  <thead>
    <th data-priority="1">Position</th>
    <th>Team</th>
    <th data-priority="2">Rating</th>
    <th data-priority="3">Points</th>
  </thead>
  <tbody>
    <tr>
      <th>1</th>
      <td>SOUTH AFRICA</td>
      <td>114</td>
      <td>3308</td>
    </tr>
    <tr>
      <th>2</th>
      <td>INDIA</td>
      <td>110</td>
      <td>3535</td>
    </tr>
    <tr>
      <th>3</th>
      <td>AUSTRALIA</td>
      <td>109</td>
      <td>4376</td>
    </tr>
    <tr>
      <th>4</th>
      <td>PAKISTAN</td>
      <td>106</td>
      <td>2977</td>
    </tr>
    <tr>
      <th>5</th>
```

```
<td>NEW ZEALAND</td>
<td>99</td>
<td>3578</td>
</tr>
<tr>
    <th>6</th>
    <td>ENGLAND</td>
    <td>99</td>
    <td>3940</td>
</tr>
<tr>
    <th>7</th>
    <td>SRI LANKA</td>
    <td>89</td>
    <td>3123</td>
</tr>
<tr>
    <th>8</th>
    <td>WEST INDIES</td>
    <td>76</td>
    <td>2504</td>
</tr>
<tr>
    <th>9</th>
    <td>BANGLADESH</td>
    <td>47</td>
    <td>1026</td>
</tr>
<tr>
    <th>10</th>
    <td>ZIMBABWE</td>
    <td>5</td>
    <td>53</td>
</tr>
</tbody>
</table>
<div data-role="footer">
```

```

<h2>Footer</h2>
</div>
</body>
</html>

```

## Output

Let's carry out the following steps to see how the above code works:

- Save the above html code as **jqm\_table\_reflow.html** file in your server root folder.
- Open this HTML file as [http://localhost/jqm\\_table\\_reflow.html](http://localhost/jqm_table_reflow.html) and the following output will be displayed.

## Header

| <b>Position</b> | <b>Team</b>  | <b>Rating Points</b> |
|-----------------|--------------|----------------------|
| <b>1</b>        | SOUTH AFRICA | 114 3308             |
| <b>2</b>        | INDIA        | 110 3535             |
| <b>3</b>        | AUSTRALIA    | 109 4376             |
| <b>4</b>        | PAKISTAN     | 106 2977             |
| <b>5</b>        | NEW ZEALAND  | 99 3578              |
| <b>6</b>        | ENGLAND      | 99 3940              |
| <b>7</b>        | SRI LANKA    | 89 3123              |
| <b>8</b>        | WEST INDIES  | 76 2504              |
| <b>9</b>        | BANGLADESH   | 47 1026              |
| <b>10</b>       | ZIMBABWE     | 5 53                 |

## Footer

## Table Reflow Heading Groups

---

### Description

Displays the column headings as sub headings between the data sets and represents the data in terms of value pairs using ***data-mode="reflow"*** attribute.

## Example

Following example demonstrates the use of table reflow heading groups in the jQuery Mobile Framework.

```
<!DOCTYPE html>
<head>
<title>Table Reflow Heading Group</title>
<meta name="viewport" content="width=device-width, initial-scale=1">
<link rel="stylesheet"
href="https://code.jquery.com/mobile/1.4.5/jquery.mobile-1.4.5.min.css">
<script src="https://code.jquery.com/jquery-1.11.3.min.js"></script>
<script src="https://code.jquery.com/mobile/1.4.5/jquery.mobile-
1.4.5.min.js"></script>
</head>
<body>
<div data-role="header">
    <h2>Header</h2>
</div>
<table data-role="table" id="table_columns" data-mode="reflow" class="table-
stroke">
    <thead>
        <tr class="th-groups">
            <th></th>
            <th colspan="3" data-priority="4">INDIA</th>
            <th colspan="3" data-priority="3">AUSTRALIA</th>
            <th colspan="3" data-priority="2">ENGLAND</th>
        </tr>
        <tr>
            <th>Stadium</th>
            <th>Played</th>
            <th>Won</th>
            <th>Loss</th>
            <th>Played</th>
            <th>Won</th>
            <th>Loss</th>
            <th>Played</th>
        </tr>
    <tbody>
        <tr>
            <td>Wankhede</td>
            <td>10</td>
            <td>5</td>
            <td>3</td>
            <td>10</td>
            <td>5</td>
            <td>3</td>
            <td>10</td>
        </tr>
        <tr>
            <td>Eden Park</td>
            <td>10</td>
            <td>6</td>
            <td>4</td>
            <td>10</td>
            <td>6</td>
            <td>4</td>
            <td>10</td>
        </tr>
        <tr>
            <td>Lord's</td>
            <td>10</td>
            <td>7</td>
            <td>3</td>
            <td>10</td>
            <td>7</td>
            <td>3</td>
            <td>10</td>
        </tr>
    </tbody>
</table>
</body>
```

```
<th>Won</th>
<th>Loss</th>
</tr>
</thead>
<tbody>
<tr>
    <th>Sydney</th>
    <td>20</td>
    <td>14</td>
    <td>6</td>
    <td>15</td>
    <td>10</td>
    <td>5</td>
    <td>10</td>
    <td>3</td>
    <td>7</td>
</tr>
<tr>
    <th>Eden Gardens</th>
    <td>10</td>
    <td>9</td>
    <td>1</td>
    <td>8</td>
    <td>3</td>
    <td>5</td>
    <td>9</td>
    <td>5</td>
    <td>4</td>
</tr>
<tr>
    <th>Oval</th>
    <td>12</td>
    <td>9</td>
    <td>3</td>
    <td>15</td>
    <td>3</td>
    <td>12</td>
```

```

<td>20</td>
<td>15</td>
<td>5</td>
</tr>
</tbody>
</table>
<div data-role="footer">
    <h2>Footer</h2>
</div>
</body>
</html>

```

## Output

Let's carry out the following steps to see how the above code works:

- Save the above html code as **jqm\_table\_reflow\_heading\_groups.html** file in your server root folder.
- Open this HTML file as [http://localhost/jqm\\_table\\_reflow\\_heading\\_groups.html](http://localhost/jqm_table_reflow_heading_groups.html) and the following output will be displayed.

## Header

|              | INDIA  |     |      | AUSTRALIA |     |      | ENGLAND |     |      |
|--------------|--------|-----|------|-----------|-----|------|---------|-----|------|
| Stadium      | Played | Won | Loss | Played    | Won | Loss | Played  | Won | Loss |
| Sydney       | 20     | 14  | 6    | 15        | 10  | 5    | 10      | 3   | 7    |
| Eden Gardens | 10     | 9   | 1    | 8         | 3   | 5    | 9       | 5   | 4    |
| Oval         | 12     | 9   | 3    | 15        | 3   | 12   | 20      | 15  | 5    |

## Footer

## Table Reflow Stripes and Strokes

### Description

You can display the data sets in stripes and strokes format using the **table-stripe** and **table-stroke** classes.

### Example

Following example demonstrates the use of table reflow stripes and strokes in the jQuery Mobile Framework.

```
<!DOCTYPE html>
<head>
<title>Table Column Toggle Customization</title>
<meta name="viewport" content="width=device-width, initial-scale=1">
<link rel="stylesheet"
href="https://code.jquery.com/mobile/1.4.5/jquery.mobile-1.4.5.min.css">
<script src="https://code.jquery.com/jquery-1.11.3.min.js"></script>
<script src="https://code.jquery.com/mobile/1.4.5/jquery.mobile-
1.4.5.min.js"></script>
</head>
<body>
<div data-role="header">
    <h2>Header</h2>
</div>
<table data-role="table" id="table_columns" data-mode="reflow" class="table-
stripe table-stroke ui-responsive">
    <thead>
        <tr>
            <th data-priority="1">Position</th>
            <th>Team</th>
            <th data-priority="2">Rating</th>
            <th data-priority="3">Points</th>
        </tr>
    </thead>
    <tbody>
        <tr>
            <th>1</th>
            <td>SOUTH AFRICA</td>
            <td>114</td>
            <td>3308</td>
        </tr>
        <tr>
            <th>2</th>
            <td>INDIA</td>
            <td>110</td>
            <td>3535</td>
        </tr>
    </tbody>
</table>

```

```
</tr>
<tr>
    <th>3</th>
    <td>AUSTRALIA</td>
    <td>109</td>
    <td>4376</td>
</tr>
<tr>
    <th>4</th>
    <td>PAKISTAN</td>
    <td>106</td>
    <td>2977</td>
</tr>
<tr>
    <th>5</th>
    <td>NEW ZEALAND</td>
    <td>99</td>
    <td>3578</td>
</tr>
<tr>
    <th>6</th>
    <td>ENGLAND</td>
    <td>99</td>
    <td>3940</td>
</tr>
<tr>
    <th>7</th>
    <td>SRI LANKA</td>
    <td>89</td>
    <td>3123</td>
</tr>
<tr>
    <th>8</th>
    <td>WEST INDIES</td>
    <td>76</td>
    <td>2504</td>
</tr>
```

```
<tr>
  <th>9</th>
  <td>BANGLADESH</td>
  <td>47</td>
  <td>1026</td>
</tr>
<tr>
  <th>10</th>
  <td>ZIMBABWE</td>
  <td>5</td>
  <td>53</td>
</tr>
</tbody>
</table>
<div data-role="footer">
  <h2>Footer</h2>
</div>
</body>
</html>
```

## Output

Let's carry out the following steps to see how the above code works:

- Save the above html code as **jqm\_table\_reflow\_stripes\_strokes.html** file in your server root folder.
- Open this HTML file as [http://localhost/jqm\\_table\\_reflow\\_stripes\\_strokes.html](http://localhost/jqm_table_reflow_stripes_strokes.html) and the following output will be displayed.

## Header

| Position | Team         | Rating | Points |
|----------|--------------|--------|--------|
| 1        | SOUTH AFRICA | 114    | 3308   |
| 2        | INDIA        | 110    | 3535   |
| 3        | AUSTRALIA    | 109    | 4376   |
| 4        | PAKISTAN     | 106    | 2977   |
| 5        | NEW ZEALAND  | 99     | 3578   |
| 6        | ENGLAND      | 99     | 3940   |
| 7        | SRI LANKA    | 89     | 3123   |
| 8        | WEST INDIES  | 76     | 2504   |
| 9        | BANGLADESH   | 47     | 1026   |
| 10       | ZIMBABWE     | 5      | 53     |

## Footer

### Table Reflow Custom Styles

#### Description

You can specify the custom styles for the reflow table.

#### Example

Following example demonstrates the use of table reflow custom styles in the jQuery Mobile Framework.

```
<!DOCTYPE html>
<head>
<title>Table Reflow Custom Styles</title>
<meta name="viewport" content="width=device-width, initial-scale=1">
<link rel="stylesheet"
href="https://code.jquery.com/mobile/1.4.5/jquery.mobile-1.4.5.min.css">
<script src="https://code.jquery.com/jquery-1.11.3.min.js"></script>
<script src="https://code.jquery.com/mobile/1.4.5/jquery.mobile-
1.4.5.min.js"></script>
<style>
#table_columns thead th{
```

```
color: #FF00BF;
font-weight: bold;
}

#table_columns tbody td{
color: #00BFFF;
font-weight: normal;
}

</style>
</head>
<body>
<div data-role="header">
<h2>Header</h2>
</div>
<table data-role="table" id="table_columns" data-mode="reflow" class="ui-responsive">
<thead>
<tr>
<th data-priority="1">Position</th>
<th>Team</th>
<th data-priority="2">Rating</th>
<th data-priority="3">Points</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>SOUTH AFRICA</td>
<td>114</td>
<td>3308</td>
</tr>
<tr>
<td>2</td>
<td>INDIA</td>
<td>110</td>
<td>3535</td>
</tr>
<tr>
```

```
<td>3</td>
<td>AUSTRALIA</td>
<td>109</td>
<td>4376</td>
</tr>
<tr>
<td>4</td>
<td>PAKISTAN</td>
<td>106</td>
<td>2977</td>
</tr>
<tr>
<td>5</td>
<td>NEW ZEALAND</td>
<td>99</td>
<td>3578</td>
</tr>
<tr>
<td>6</td>
<td>ENGLAND</td>
<td>99</td>
<td>3940</td>
</tr>
<tr>
<td>7</td>
<td>SRI LANKA</td>
<td>89</td>
<td>3123</td>
</tr>
<tr>
<td>8</td>
<td>WEST INDIES</td>
<td>76</td>
<td>2504</td>
</tr>
<tr>
<td>9</td>
```

```
<td>BANGLADESH</td>
<td>47</td>
<td>1026</td>
</tr>
<tr>
    <td>10</td>
    <td>ZIMBABWE</td>
    <td>5</td>
    <td>53</td>
</tr>
</tbody>
</table>
<div data-role="footer">
    <h2>Footer</h2>
</div>
</body>
</html>
```

## Output

Let's carry out the following steps to see how the above code works:

- Save the above html code as **jqm\_table\_reflow\_custom\_styles.html** file in your server root folder.
- Open this HTML file as [http://localhost/jqm\\_table\\_reflow\\_custom\\_styles.html](http://localhost/jqm_table_reflow_custom_styles.html) and the following output will be displayed.

## Header

| <b>Position</b> | <b>Team</b>  | <b>Rating</b> | <b>Points</b> |
|-----------------|--------------|---------------|---------------|
| 1               | SOUTH AFRICA | 114           | 3308          |
| 2               | INDIA        | 110           | 3535          |
| 3               | AUSTRALIA    | 109           | 4376          |
| 4               | PAKISTAN     | 106           | 2977          |
| 5               | NEW ZEALAND  | 99            | 3578          |
| 6               | ENGLAND      | 99            | 3940          |
| 7               | SRI LANKA    | 89            | 3123          |
| 8               | WEST INDIES  | 76            | 2504          |
| 9               | BANGLADESH   | 47            | 1026          |
| 10              | ZIMBABWE     | 5             | 53            |

## Footer

### Tabs Widget

#### Description

The tabs widget is jQuery ui tabs widget's extension, which accepts all the methods and options.

Following table lists different types of tabs used in jQuery Mobile.

| <b>Sr. No.</b> | <b>Tabs Types &amp; Description</b>  |
|----------------|--|
| 1              | <u><b>Use navbar for tabs</b></u><br>To use navbar for your tabs, you can declare navbar inside your tabs. |
| 2              | <u><b>Tabbed page content</b></u><br>You can swap out the entire page contents using the tabs.             |

## Use Navbar for Tabs

### Description

To use navbar for your tabs, you can declare the navbar inside your tabs.

### Example

Following example demonstrates the use of navbar for tabs in jQuery Mobile.

```
<!DOCTYPE html>
<html>
<head>
    <title>Use navbar for tabs</title>
    <meta name="viewport" content="width=device-width, initial-scale=1">
    <link rel="stylesheet"
    href="https://code.jquery.com/mobile/1.4.5/jquery.mobile-1.4.5.min.css">
    <script src="https://code.jquery.com/jquery-1.11.3.min.js"></script>
    <script src="https://code.jquery.com/mobile/1.4.5/jquery.mobile-
1.4.5.min.js"></script>
    <style>
        .tablist-left {width: 25%; display: inline-block; }
        .tablist-content {width: 60%; display: inline-block; vertical-align: top;
margin-left: 5%; }
    </style>
</head>
<body>
    <div data-role="tabs" id="tabs">
        <div data-role="navbar">
            <ul>
                <li><a href="#one" data-ajax="false">Tab one</a></li>
                <li><a href="#two" data-ajax="false">Tab two</a></li>
                <li><a href="/jquery_mobile/src/tabs_navbars2.html" data-
ajax="false">Tab three</a></li>
            </ul>
        </div>
        <div id="one" class="ui-body-d ui-content">
            <h1>This is first tab contents.</h1>
            <p>Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do
eiusmod tempor incididunt ut labore et dolore magna aliqua. Ut enim ad minim
veniam, quis nostrud exercitation ullamco laboris nisi ut aliquip ex ea commodo
consequat</p>
    </div>
</div>
```

```

</div>
<div id="two">
    <ul data-role="listview" data-inset="true">
        <li><a href="#">Audi</a></li>
        <li><a href="#">Hyundai</a></li>
        <li><a href="#">Honda</a></li>
    </ul>
</div>
</div>

<h2>Use inset listview for tabs</h2>
<div data-role="tabs">
    <ul data-role="listview" data-inset="true" class="tablist-left">
        <li><a href="#one" data-ajax="false">one</a></li>
        <li><a href="#two" data-ajax="false">two</a></li>
        <li><a href="tabs_navbars2.html" data-ajax="false">three</a></li>
    </ul>
    <div id="one" class="ui-body-d tablist-content">
        <h1>First tab contents</h1>
    </div>
    <ul id="two" class="tablist-content" data-role="listview" data-
inset="true">
        <li><a href="#">Audi</a></li>
        <li><a href="#">Hyundai</a></li>
        <li><a href="#">Honda</a></li>
    </ul>
</div>
</body>
</html>

```

## Output

Let's carry out the following steps to see how the above code works:

- Save the above html code as **tabs\_navbars.html** file in your server root folder.
- Open this HTML file as [http://localhost/tabs\\_navbars.html](http://localhost/tabs_navbars.html) and the following output will be displayed.

- [Tab one](#)
- [Tab two](#)
- [Tab three](#)

## This is first tab contents.

*Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua. Ut enim ad minim veniam, quis nostrud exercitation ullamco laboris nisi ut aliquip ex ea commodo consequat*

- [Audi](#)
- [Hyundai](#)
- [Honda](#)

## Use inset listview for tabs

- [one](#)
- [two](#)
- [three](#)

### First tab contents

- [Audi](#)
- [Hyundai](#)
- [Honda](#)

## Tabbed Page Content

---

### Description

You can swap out the entire page contents using the tabs.

### Example

Following example demonstrates the use of tabbed page content in jQuery Mobile.

```
<!DOCTYPE html>
<html>
<head>
  <title>Use navbar for tabs</title>
  <meta name="viewport" content="width=device-width, initial-scale=1">
  <link rel="stylesheet"
  href="https://code.jquery.com/mobile/1.4.5/jquery.mobile-1.4.5.min.css">
```

```
<link rel="stylesheet"
href="http://demos.jquerymobile.com/1.4.5/_assets/css/jqm-demos.css">
<script src="https://code.jquery.com/jquery-1.11.3.min.js"></script>
<script src="https://code.jquery.com/mobile/1.4.5/jquery.mobile-
1.4.5.min.js"></script>
</head>
<body>
<div data-role="page" class="jqm-demos" data-quicklinks="true">
<div data-demo-html="true">
<div data-role="tabs">
<div data-role="navbar">
<ul>
<li><a href="#one" data-theme="a" data-ajax="false">Tab one</a></li>
<li><a href="#two" data-theme="a" data-ajax="false">Tab two</a></li>
<li><a href="/jquery_mobile/src/tabs_navbars2.html" data-
theme="a" data-ajax="false">Tab three</a></li>
</ul>
</div>

<div id="one" class="ui-content">
<h1>This is first tab content</h1>
</div>
<div id="two" class="ui-content">
<ul data-role="listview">
<li><a href="#">Audi</a></li>
<li><a href="#">Hyundai</a></li>
<li><a href="#">Honda</a></li>
</ul>
</div>
</div>
<div data-role="footer" data-position="fixed" data-tap-toggle="false"
class="jqm-footer">
<p>Tutorials point jQuery Mobile<span class="jqm-
version"></span></p>
<p>Copyright 2016 Tutorials point</p>
</div>
</div>
</div>
```

```
</body>
</html>
```

## Output

Let's carry out the following steps to see how the above code works:

- Save the above html code as **tabs\_page\_content.html** file in your server root folder.
- Open this HTML file as [http://localhost/tabs\\_page\\_content.html](http://localhost/tabs_page_content.html) and the following output will be displayed.

- [Tab one](#)
- [Tab two](#)
- [Tab three](#)

## This is first tab content

- [Audi](#)
- [Hyundai](#)
- [Honda](#)

Tutorialspoint jQuery Mobile

Copyright 2016 Tutorialspoint

## Textinput Widget

---

### Description

The `<input>` tag is used to declare an input element, a control that allows the user to input data.

Following table demonstrates the types of textinput widget in detail.

| Sr. No. | Types & Description  |
|---------|--|
| 1       | <b><u>Text and Textarea</u></b><br>Text and textarea allows the user to input text for single and multiple line. |
| 2       | <b><u>Search</u></b><br>Search input type helps the user to find something from a particular page.               |

|    |   |
|----|---|
| 3  | <b><u>Number</u></b><br>Number type input allows the user to input number type character in the block.            |
| 4  | <b><u>Date, Month and Week</u></b><br>It allows the user to input date, month, and week.                          |
| 5  | <b><u>Time and Color</u></b><br>Allows to input the time and select color from the displayed box.                 |
| 6  | <b><u>Telephone and Email</u></b><br><i>tel</i> type input allows the user to input the telephone number.         |
| 7  | <b><u>URL and Password</u></b><br>The url is used to define the absolute url's.                                   |
| 8  | <b><u>Datetime</u></b><br><i>datetime</i> type attribute in input defines the date and time control.              |
| 9  | <b><u>File</u></b><br>The <i>file</i> type attribute allows the users to upload files to the server of MIME type. |
| 10 | <b><u>Mini</u></b><br>Include attribute <i>data-mini="true"</i> for creating a mini version textinput.            |
| 11 | <b><u>Placeholder</u></b><br>Placeholder specifies a short hint that describes the expected value.                |
| 12 | <b><u>Value and Text</u></b><br><i>value=""</i> attribute in input provides an initial value inside the control.  |
| 13 | <b><u>Label Hidden</u></b><br><i>ui-hidden-accessible</i> class is used to hide the the label.                    |
| 14 | <b><u>Disabled</u></b><br>Attribute <i>disabled="disabled"</i> is used to disable the input control.              |
| 15 | <b><u>Enhanced</u></b><br><i>data-enhanced="true"</i> attribute in the input field allows to enhance the input.   |
| 16 | <b><u>Fieldcontain</u></b><br><i>ui-field-contain</i> class is used to wrap the label and input.                  |
| 17 | <b><u>Fieldcontain, mini size</u></b>   |

|  |   |
|--|---|
|  | The wrapped label and input are created in mini version using the attribute <code>data-mini="true"</code> . |
|--|---|

## Textinput Text and Textarea

### Description

- Text type input allows the user to input one-line text.
- The `<textarea>` tag allows the user to input text over multiple rows.

### Example

Following example demonstrates the use of text and textarea input in the jQuery Mobile.

```
<!DOCTYPE html>
<html>
<head>
    <meta name="viewport" content="width=device-width, initial-scale=1">
    <link rel="stylesheet"
    href="https://code.jquery.com/mobile/1.4.5/jquery.mobile-1.4.5.min.css">
    <script src="https://code.jquery.com/jquery-1.11.3.min.js"></script>
    <script src="https://code.jquery.com/mobile/1.4.5/jquery.mobile-
    1.4.5.min.js"></script>
</head>
<body>
<form>
    <h2>Text Input Example</h2>
    <label for="text">Text input</label>
    <input type="text" id="text" data-clear-btn="false">
    <label for="text1">Clear the Text</label>
    <input type="text" data-clear-btn="true" id="text1">

    <h2>Textarea Example</h2>
    <textarea type="textarea" id="txt"></textarea>
</form>
</body>
</html>
```

## Output

Let's carry out the following steps to see how the above code works:

- Save the above html code as **textinput\_text.html** file in your server root folder.
- Open this HTML file as [http://localhost/textinput\\_text.html](http://localhost/textinput_text.html) and the following output will be displayed.

### Text Input Example

Text input  Clear the Text

### Textarea Example

## Textinput Search

---

### Description

Search input type helps the user to find something from a particular page.

### Example

Following example demonstrates the use of search input in the jQuery Mobile.

```
<!DOCTYPE html>
<html>
<head>
    <meta name="viewport" content="width=device-width, initial-scale=1">
    <link rel="stylesheet"
    href="https://code.jquery.com/mobile/1.4.5/jquery.mobile-1.4.5.min.css">
    <script src="https://code.jquery.com/jquery-1.11.3.min.js"></script>
    <script src="https://code.jquery.com/mobile/1.4.5/jquery.mobile-
1.4.5.min.js"></script>
</head>
<body>
<form>
    <h2>Search Input Example</h2>
    <input type="search" id="src">
```

```
</form>
</body>
</html>
```

## Output

Let's carry out the following steps to see how the above code works:

- Save the above html code as **textinput\_search.html** file in your server root folder.
- Open this HTML file as [http://localhost/textinput\\_search.html](http://localhost/textinput_search.html) and the following output will be displayed.

## Search Input Example

## Textinput Number

### Description

Number input type allows the user to input numerical values.

### Example

Following example demonstrates the use of number input in the jQuery Mobile.

```
<!DOCTYPE html>
<html>
<head>
    <meta name="viewport" content="width=device-width, initial-scale=1">
    <link rel="stylesheet"
    href="https://code.jquery.com/mobile/1.4.5/jquery.mobile-1.4.5.min.css">
    <script src="https://code.jquery.com/jquery-1.11.3.min.js"></script>
    <script src="https://code.jquery.com/mobile/1.4.5/jquery.mobile-
    1.4.5.min.js"></script>
</head>
```

```

<body>
<form>
  <h2>Number Input Example</h2>
  <label for="number">Simple Number Input</label>
  <input type="number" id="number" data-clear-btn="false">
  <label for="number1">Clear the Number Field</label>
  <input type="number" data-clear-btn="true" id="number1">
  <label for="number2">Number Pattern</label>
  <input type="number" id="number2" pattern="[0-9]*" data-clear-btn="false">
  <label for="number3">Clear the Number Pattern Field</label>
  <input type="number" data-clear-btn="true" id="number3" pattern="[0-9]*">
</form>
</body>
</html>

```

## Output

Let's carry out the following steps to see how the above code works:

- Save the above html code as **textinput\_number.html** file in your server root folder.
- Open this HTML file as [http://localhost/textinput\\_number.html](http://localhost/textinput_number.html) and the following output will be displayed.

## Number Input Example

|                     |                               |                        |
|---------------------|-------------------------------|------------------------|
| Simple Number Input | <input type="text" value=""/> | Clear the Number Field |
| Number Pattern      | <input type="text" value=""/> | Clear the Number       |
| Pattern Field       | <input type="text" value=""/> |                        |

## Textinput Date, Month, and Week

---

### Description

**Date input type** attribute allows the user to input date (year, month, day) encoded according to ISO 8601.

**Month input type** allows the user to select date consisting of a year and a month encoded according to ISO 8601.

**Week input** allows the user to select date consisting of a year and a week number encoded according to ISO 8601.

### Example

Following example demonstrates the use of date, month, and week input in the jQuery Mobile.

```
<!DOCTYPE html>
<html>
<head>
    <meta name="viewport" content="width=device-width, initial-scale=1">
    <link rel="stylesheet"
    href="https://code.jquery.com/mobile/1.4.5/jquery.mobile-1.4.5.min.css">
    <script src="https://code.jquery.com/jquery-1.11.3.min.js"></script>
    <script src="https://code.jquery.com/mobile/1.4.5/jquery.mobile-
1.4.5.min.js"></script>
</head>
<body>
<form>

    <h2>Date Input Example</h2>
    <label for="src">Date Input</label>
    <input type="date" id="src" data-clear-btn="false">
    <label for="src1">Clear the Date input Field</label>
    <input type="date" data-clear-btn="true" id="src1">

    <h2>Month Input Example</h2>
    <label for="src2">Month Input</label>
    <input type="month" id="src2" data-clear-btn="false">
    <label for="src3">Clear the Month input Field</label>
    <input type="month" data-clear-btn="true" id="src3">

    <h2>Week Input Example</h2>

```

```

<label for="src4">Week Input</label>
<input type="week" id="src4" data-clear-btn="false">
<label for="src5">Clear the Week input Field</label>
<input type="week" data-clear-btn="true" id="src5">
</form>
</body>
</html>

```

## Output

Let's carry out the following steps to see how the above code works:

- Save the above html code as **textinput\_date\_month\_week.html** file in your server root folder.
- Open this HTML file as [http://localhost/textinput\\_date\\_month\\_week.html](http://localhost/textinput_date_month_week.html) and the following output will be displayed.

## Date Input Example

Date Input  Clear the Date input Field

## Month Input Example

Month Input  Clear the Month input Field

## Week Input Example

Week Input  Clear the Week input Field

## Textinput Time and Color

### Description

- Time input type allows the user to input time (hour, minute, seconds, fractional seconds) encoded according to ISO 8601.
- Color input type allows the user to select a particular color from the displayed color box.

## Example

Following example demonstrates the use of time and color input in the jQuery Mobile.

```
<!DOCTYPE html>
<html>
<head>
    <meta name="viewport" content="width=device-width, initial-scale=1">
    <link rel="stylesheet"
    href="https://code.jquery.com/mobile/1.4.5/jquery.mobile-1.4.5.min.css">
    <script src="https://code.jquery.com/jquery-1.11.3.min.js"></script>
    <script src="https://code.jquery.com/mobile/1.4.5/jquery.mobile-
    1.4.5.min.js"></script>
</head>
<body>
<form>
    <h2>Time Input Example</h2>
    <label for="src">Time Input</label>
    <input type="time" id="src" data-clear-btn="false">
    <label for="src1">Clear the Time Field</label>
    <input type="time" data-clear-btn="true" id="src1">

    <h2>Color Input Example</h2>
    <label for="src2">Color Input</label>
    <input type="color" id="src2" data-clear-btn="false">
    <label for="src3">Clear the Color Field</label>
    <input type="color" data-clear-btn="true" id="src3">
</form>
</body>
</html>
```

## Output

Let's carry out the following steps to see how the above code works:

- Save the above html code as **textinput\_time\_color.html** file in your server root folder.
- Open this HTML file as [http://localhost/textinput\\_time\\_color.html](http://localhost/textinput_time_color.html) and the following output will be displayed.

## Time Input Example

Time Input  Clear the Time Field

## Color Input Example

Color Input  Clear the Color Field

## Textinput Telephone and Email

---

### Description

**tel type** input allows the user to input the telephone number.

**email type** attribute in input accepts only email value. This type is used for input fields that should contain an e-mail address.

### Example

Following example demonstrates the use of telephone and email input in the jQuery Mobile.

```
<!DOCTYPE html>
<html>
<head>
    <meta name="viewport" content="width=device-width, initial-scale=1">
    <link rel="stylesheet"
    href="https://code.jquery.com/mobile/1.4.5/jquery.mobile-1.4.5.min.css">
    <script src="https://code.jquery.com/jquery-1.11.3.min.js"></script>
    <script src="https://code.jquery.com/mobile/1.4.5/jquery.mobile-
    1.4.5.min.js"></script>
</head>
<body>
<form>
    <h2>Telephone Input Example</h2>
    <label for="src">Telephone Input</label>
```

```

<input type="tel" id="src" data-clear-btn="false">
<label for="src1">Clear the Telephone Field</label>
<input type="tel" data-clear-btn="true" id="src1">

<h2>Email Input Example</h2>
<label for="src2">Email Input</label>
<input type="email" id="src2" data-clear-btn="false">
<label for="src3">Clear the Email Field</label>
<input type="email" data-clear-btn="true" id="src3">
</form>
</body>
</html>

```

## Output

Let's carry out the following steps to see how the above code works:

- Save the above html code as **textinput\_telephone\_email.html** file in your server root folder.
- Open this HTML file as [http://localhost/textinput\\_telephone\\_email.html](http://localhost/textinput_telephone_email.html) and the following output will be displayed.

## Telephone Input Example

Telephone Input  Clear the Telephone Field



## Email Input Example

Email Input  Clear the Email Field

## Textinput URL and Password

### Description

**url** **input** type accepts only URL value. This type is used for input fields that should contain a URL address.

**password input** type is used to create a hidden text control, it is a field for sensitive information.

## Example

Following example demonstrates the use of URL and password input in the jQuery Mobile.

```
<!DOCTYPE html>
<html>
<head>
    <meta name="viewport" content="width=device-width, initial-scale=1">
    <link rel="stylesheet"
    href="https://code.jquery.com/mobile/1.4.5/jquery.mobile-1.4.5.min.css">
    <script src="https://code.jquery.com/jquery-1.11.3.min.js"></script>
    <script src="https://code.jquery.com/mobile/1.4.5/jquery.mobile-
1.4.5.min.js"></script>
</head>
<body>
<form>
    <h2>URL Input Example</h2>
    <label for="src">URL Input</label>
    <input type="url" id="src" data-clear-btn="false">
    <label for="src1">Clear the URL Field</label>
    <input type="url" data-clear-btn="true" id="src1">

    <h2>Password Input Example</h2>
    <label for="src2">Password Input</label>
    <input type="password" id="src2" data-clear-btn="false">
    <label for="src3">Clear the Password Field</label>
    <input type="password" data-clear-btn="true" id="src3">
</form>
</body>
</html>
```

## Output

Let's carry out the following steps to see how the above code works:

- Save the above html code as **textinput\_url\_password.html** file in your server root folder.
- Open this HTML file as [http://localhost/textinput\\_url\\_password.html](http://localhost/textinput_url_password.html) and the following output will be displayed.

## URL Input Example

URL Input  Clear the URL Field

## Password Input Example

Password Input  Clear the Password Field

## Textinput Datetime

---

### Description

**datetime** type attribute in input defines the date and time control. A date and time (year, month, day, hour, minute, second, fractions of a second) is encoded according to ISO 8601 with the time zone set to UTC.

### Example

Following example demonstrates the use of datetime input in the jQuery Mobile.

```
<!DOCTYPE html>
<html>
<head>
    <meta name="viewport" content="width=device-width, initial-scale=1">
    <link rel="stylesheet"
    href="https://code.jquery.com/mobile/1.4.5/jquery.mobile-1.4.5.min.css">
    <script src="https://code.jquery.com/jquery-1.11.3.min.js"></script>
    <script src="https://code.jquery.com/mobile/1.4.5/jquery.mobile-
    1.4.5.min.js"></script>
</head>
<body>
<form>
    <h2>Datetime Input Example</h2>
    <label for="src">Datetime Input</label>
    <input type="datetime" id="src" data-clear-btn="false">

    <label for="src1">Clear the Datetime Field</label>
    <input type="datetime" data-clear-btn="true" id="src1">

```

```

<label for="src2">Datetime Local Input</label>
<input type="datetime-local" id="src2" data-clear-btn="false">

<label for="src3">Clear the Datetime Local Field</label>
<input type="datetime-local" data-clear-btn="true" id="src3">
</form>
</body>
</html>

```

## Output

Let's carry out the following steps to see how the above code works:

- Save the above html code as **textinput\_datetime.html** file in your server root folder.
- Open this HTML file as [http://localhost/textinput\\_datetime.html](http://localhost/textinput_datetime.html) and the following output will be displayed.

## Datetime Input Example

Datetime Input  Clear the Datetime Field

Datetime Local Input  dd----- yyyy -- : -- -- Clear

the Datetime Local Field  dd----- yyyy -- : -- --

## Textinput File

---

### Description

The **file** type attribute allows the users to upload files to the server of MIME type. An uploaded file could be a text file or image file or any document.

### Example

Following example demonstrates the use of file input in the jQuery Mobile.

```

<!DOCTYPE html>
<html>
<head>
<meta name="viewport" content="width=device-width, initial-scale=1">

```

```

<link rel="stylesheet"
href="https://code.jquery.com/mobile/1.4.5/jquery.mobile-1.4.5.min.css">
<script src="https://code.jquery.com/jquery-1.11.3.min.js"></script>
<script src="https://code.jquery.com/mobile/1.4.5/jquery.mobile-
1.4.5.min.js"></script>
</head>
<body>
<form>
  <h2>File Input Example</h2>
  <label for="src">File Input</label>
  <input type="file" id="src" data-clear-btn="false">

  <label for="src1">Clear the File Field</label>
  <input type="file" data-clear-btn="true" id="src1">
</form>
</body>
</html>

```

## Output

Let's carry out the following steps to see how the above code works:

- Save the above html code as **textinput\_file.html** file in your server root folder.
- Open this HTML file as [http://localhost/textinput\\_file.html](http://localhost/textinput_file.html) and the following output will be displayed.

## File Input Example

No file chosen     

## Textinput Mini

### Description

Include attribute ***data-mini="true"*** for creating a mini version textinput.

## Example

Following example demonstrates the use of mini input in the jQuery Mobile.

```
<!DOCTYPE html>
<html>
<head>
    <meta name="viewport" content="width=device-width, initial-scale=1">
    <link rel="stylesheet"
    href="https://code.jquery.com/mobile/1.4.5/jquery.mobile-1.4.5.min.css">
    <script src="https://code.jquery.com/jquery-1.11.3.min.js"></script>
    <script src="https://code.jquery.com/mobile/1.4.5/jquery.mobile-
1.4.5.min.js"></script>
</head>
<body>
<form>
    <h2>Text mini Input Example</h2>
    <label for="src">Text Input</label>
    <input type="text" id="src" data-clear-btn="false" data-mini="true">

    <label for="src1">Clear the Text mini Field</label>
    <input type="text" data-clear-btn="true" id="src1" data-mini="true">

    <label for="src2">Search mini Input</label>
    <input type="search" id="src2" data-mini="true">

    <label for="src3">Textarea mini Field</label>
    <textarea type="textarea" id="src3" data-mini="true"></textarea>
</form>
</body>
</html>
```

## Output

Let's carry out the following steps to see how the above code works:

- Save the above html code as ***textinput\_mini.html*** file in your server root folder.
- Open this HTML file as [http://localhost/textinput\\_mini.html](http://localhost/textinput_mini.html) and the following output will be displayed.

## Text mini Input Example

Text Input  Clear the Text mini Field 
  
 Search mini Input  Textarea mini Field

## Textinput Placeholder

### Description

Placeholder specifies a short hint that describes the expected value. Add **attribute *placeholder=" "*** in the input element to write the placeholder.

### Example

Following example demonstrates the use of placeholder input in the jQuery Mobile.

```

<!DOCTYPE html>
<html>
<head>
    <meta name="viewport" content="width=device-width, initial-scale=1">
    <link rel="stylesheet"
    href="https://code.jquery.com/mobile/1.4.5/jquery.mobile-1.4.5.min.css">
    <script src="https://code.jquery.com/jquery-1.11.3.min.js"></script>
    <script src="https://code.jquery.com/mobile/1.4.5/jquery.mobile-
1.4.5.min.js"></script>
</head>
<body>
<form>
    <h2>Placeholder Input Type Example</h2>
    <label for="src">Text Placeholder Input</label>
    <input type="text" id="src" placeholder="Enter Something">

    <label for="src1">Clear the Text Placeholder Field</label>
    <input type="text" data-clear-btn="true" id="src1" placeholder="Enter
Something">

```

```

<label for="src2">Search Placeholder Input</label>
<input type="search" id="src2" placeholder="Search">

<label for="src3"> Textarea Placeholder Field</label>
<textarea type="textarea" id="src3" placeholder="Enter your
message"></textarea>
</form>
</body>
</html>

```

## Output

Let's carry out the following steps to see how the above code works:

- Save the above html code as **textinput\_placeholder.html** file in your server root folder.
- Open this HTML file as [http://localhost/textinput\\_placeholder.html](http://localhost/textinput_placeholder.html) and the following output will be displayed.

## Placeholder Input Type Example

The screenshot shows a user interface with four input fields. From left to right: 1. A text input field with the placeholder "Enter Something". 2. A search input field with the placeholder "Search Placeholder Input". 3. A text input field with the placeholder "Placeholder Field". 4. A textarea with the placeholder "Textarea". All fields are styled consistently with jQuery Mobile's design.

## Textinput Value/Text

---

### Description

**value=""** attribute in input provides an initial value inside the control.

### Example

Following example demonstrates the use of value/text input in the jQuery Mobile.

```

<!DOCTYPE html>
<html>
<head>

```

```

<meta name="viewport" content="width=device-width, initial-scale=1">
<link rel="stylesheet"
href="https://code.jquery.com/mobile/1.4.5/jquery.mobile-1.4.5.min.css">
<script src="https://code.jquery.com/jquery-1.11.3.min.js"></script>
<script src="https://code.jquery.com/mobile/1.4.5/jquery.mobile-
1.4.5.min.js"></script>
</head>
<body>
<form>
  <h2>Value/Text Input Type Example</h2>
  <label for="src">Value/Text Input</label>
  <input type="text" id="src" value="John">

  <label for="src1">Clear the Value/Text Field</label>
  <input type="text" data-clear-btn="true" id="src1" value="John">

  <label for="src2">Search Input Value</label>
  <input type="search" id="src2" value="Search">

  <label for="src3">Textarea Value/Text </label>
  <textarea type="textarea" id="src3">Textarea</textarea>
</form>
</body>
</html>

```

## Output

Let's carry out the following steps to see how the above code works:

- Save the above html code as **textinput\_value\_text.html** file in your server root folder.
- Open this HTML file as [http://localhost/textinput\\_value\\_text.html](http://localhost/textinput_value_text.html) and the following output will be displayed.

## Value/Text Input Type Example

Value/Text Input

## Textinput Label Hidden

### Description

**ui-hidden-accessible** class is used to hide the the label.

### Example

Following example demonstrates the use of label hidden in the jQuery Mobile.

```
<!DOCTYPE html>
<html>
<head>
    <meta name="viewport" content="width=device-width, initial-scale=1">
    <link rel="stylesheet"
    href="https://code.jquery.com/mobile/1.4.5/jquery.mobile-1.4.5.min.css">
    <script src="https://code.jquery.com/jquery-1.11.3.min.js"></script>
    <script src="https://code.jquery.com/mobile/1.4.5/jquery.mobile-
1.4.5.min.js"></script>
</head>
<body>
<form>
    <h2>Label hidden Example</h2>
    <label for="src" class="ui-hidden-accessible">Label Hidden Text Input</label>
    <input type="text" id="src">

    <label for="src1" class="ui-hidden-accessible">Clear the label hidden text
    Field</label>
    <input type="text" data-clear-btn="true" id="src1">

    <label for="src2" class="ui-hidden-accessible">Label Hidden Search Input</label>

```

```

<input type="search" id="src2">

    <label for="src3" class="ui-hidden-accessible">Label Hidden Textarea
</label>
    <textarea type="textarea" id="src3"></textarea>
</form>
</body>
</html>

```

## Output

Let's carry out the following steps to see how the above code works:

- Save the above html code as **textinput\_label\_hidden.html** file in your server root folder.
- Open this HTML file as [http://localhost/textinput\\_label\\_hidden.html](http://localhost/textinput_label_hidden.html) and the following output will be displayed.

## Label hidden Example

Label Hidden Text Input  Clear the label hidden text Field  
 Label Hidden Search Input  Label  
 Hidden Textarea

## Textinput Disabled

---

### Description

Attribute **disabled="disabled"** is used to disable the input control. The button won't accept the changes from the user.

### Example

Following example demonstrates the use to disable input in the jQuery Mobile.

```

<!DOCTYPE html>
<html>
<head>

```

```
<meta name="viewport" content="width=device-width, initial-scale=1">
<link rel="stylesheet"
href="https://code.jquery.com/mobile/1.4.5/jquery.mobile-1.4.5.min.css">
<script src="https://code.jquery.com/jquery-1.11.3.min.js"></script>
<script src="https://code.jquery.com/mobile/1.4.5/jquery.mobile-
1.4.5.min.js"></script>
</head>
<body>
<form>
    <h2>Disabled Input Type Example</h2>
    <label for="src">Disabled Text Input</label>
    <input type="text" id="src" disabled="disabled">

    <label for="src1">Clear the Disabled Field</label>
    <input type="text" data-clear-btn="true" id="src1" disabled="disabled">

    <label for="src2">Disabled Search Input</label>
    <input type="search" id="src2" disabled="disabled">

    <label for="src3"> Disabled Textarea</label>
    <textarea type="textarea" id="src3" disabled="disabled"></textarea>
</form>
</body>
</html>
```

## Output

Let's carry out the following steps to see how the above code works:

- Save the above html code as **textinput\_disabled.html** file in your server root folder.
- Open this HTML file as [http://localhost/textinput\\_disabled.html](http://localhost/textinput_disabled.html) and the following output will be displayed.

## Disabled Input Type Example

Disabled Text Input  Clear the Disabled Field

Disabled Search Input  Disabled

Textarea

## Textinput Enhanced

### Description

You can increase the quality and functionality of text inputs by setting the ***data-enhanced="true"*** attribute in the input field.

### Example

Following example demonstrates the use of textinput enhanced in the jQuery Mobile.

```
<!DOCTYPE html>
<head>
    <meta name="viewport" content="width=device-width, initial-scale=1">
    <link rel="stylesheet"
    href="https://code.jquery.com/mobile/1.4.5/jquery.mobile-1.4.5.min.css">
    <script src="https://code.jquery.com/jquery-1.11.3.min.js"></script>
    <script src="https://code.jquery.com/mobile/1.4.5/jquery.mobile-
1.4.5.min.js"></script>
</head>
<body>
    <form>
        <h2>Enhanced Input Type Example</h2>
        <label for="src">Enhanced Input</label>
        <div class="ui-input-text ui-body-inherit">
            <input type="text" id="src" data-enhanced="true">
        </div>
        <label for="src2">Enhanced Search Input</label>
        <div class="ui-input-search ui-input-text ui-body-inherit">
            <input type="text" data-type="search" id="src2" data-enhanced="true">
        </div>
    </form>

```

```

        <label for="src3">Textarea Enhanced </label>
        <textarea type="text" id="src3" data-enhanced="true" class="ui-input-text ui-body-inherit ui-corner-all ui-shadow-inset ui-input-has-clear"></textarea>
    </form>
</body>
</html>

```

## Output

Let's carry out the following steps to see how the above code works:

- Save the above html code as **textinput\_enhanced.html** file in your server root folder.
- Open this HTML file as [http://localhost/textinput\\_enhanced.html](http://localhost/textinput_enhanced.html) and the following output will be displayed.

## Enhanced Input Type Example

[Enhanced Input](#)

[Enhanced Search Input](#)

[Textarea Enhanced](#)

## Textinput Fieldcontain

### Description

**ui-field-contain** class is used to wrap the label and input.

### Example

Following example demonstrates the use of fieldcontain in the jQuery Mobile.

```

<!DOCTYPE html>
<html>
<head>
    <meta name="viewport" content="width=device-width, initial-scale=1">
    <link rel="stylesheet"
    href="https://code.jquery.com/mobile/1.4.5/jquery.mobile-1.4.5.min.css">

```

```

<script src="https://code.jquery.com/jquery-1.11.3.min.js"></script>
<script src="https://code.jquery.com/mobile/1.4.5/jquery.mobile-
1.4.5.min.js"></script>
</head>
<body>
<form>

    <h2>Fieldcontain Example</h2>
    <div class="ui-field-contain">
        <label for="src">Fieldcontain Text Input</label>
        <input type="text" id="src">
    </div>

    <div class="ui-field-contain">
        <label for="src1">Clear the Fieldcontain text Input</label>
        <input type="text" data-clear-btn="true" id="src1">
    </div>

    <div class="ui-field-contain">
        <label for="src2">Fieldcontain Search Input</label>
        <input type="search" id="src2" >
    </div>

    <div class="ui-field-contain">
        <label for="src3"> Fieldcontain Textarea</label>
        <textarea type="textarea" id="src3"></textarea>
    </div>
</form>
</body>
</html>

```

## Output

Let's carry out the following steps to see how the above code works:

- Save the above html code as **textinput\_fieldcontain.html** file in your server root folder.
- Open this HTML file as [http://localhost/textinput\\_fieldcontain.html](http://localhost/textinput_fieldcontain.html) and the following output will be displayed.

## Fieldcontain Example

Fieldcontain Text Input

Clear the Fieldcontain text Input

Fieldcontain Search Input

Fieldcontain Textarea

## Textinput Fieldcontain, Mini Size

### Description

The wrapped label and input creates a mini version using the attribute ***data-mini="true"***.

### Example

Following example demonstrates the use of fieldcontain, mini size in the jQuery Mobile.

```
<!DOCTYPE html>
<html>
<head>
    <meta name="viewport" content="width=device-width, initial-scale=1">
    <link rel="stylesheet"
    href="https://code.jquery.com/mobile/1.4.5/jquery.mobile-1.4.5.min.css">
    <script src="https://code.jquery.com/jquery-1.11.3.min.js"></script>
    <script src="https://code.jquery.com/mobile/1.4.5/jquery.mobile-
    1.4.5.min.js"></script>
</head>
<body>
<form>
    <h2>Fieldcontain Example</h2>
    <div class="ui-field-contain">
        <label for="src">Fieldcontain Text Input</label>
        <input type="text" id="src" data-mini="true">
    </div>
</form>
```

```

<div class="ui-field-contain">
    <label for="src1">Clear the Fieldcontain text Input</label>
    <input type="text" data-clear-btn="true" id="src1" data-mini="true">
</div>

<div class="ui-field-contain">
    <label for="src2">Fieldcontain Search Input</label>
    <input type="search" id="src2" data-mini="true">
</div>

<div class="ui-field-contain">
    <label for="src3"> Fieldcontain Textarea</label>
    <textarea type="text" id="src3" data-mini="true"></textarea>
</div>
</form>
</body>
</html>

```

## Output

Let's carry out the following steps to see how the above code works:

- Save the above html code as **textinput\_fieldcontain\_mini.html** file in your server root folder.
- Open this HTML file as [http://localhost/textinput\\_fieldcontain\\_mini.html](http://localhost/textinput_fieldcontain_mini.html) and the following output will be displayed.

## Fieldcontain Example

Fieldcontain Text Input

Clear the Fieldcontain text Input

Fieldcontain Search Input

Fieldcontain Textarea

## Toolbar Widget

---

### Description

The jQuery mobile toolbar widget allows you to create headers and footers. Following table lists different types of toolbars.

| Sr. No. | Toolbar Types & Description   |
|---------|---|
| 1       | <b><u>Toolbar</u></b><br>To enhance both headers and footers, you can use <i>toolbar</i> widget.  |
| 2       | <b><u>Dynamic toolbars</u></b><br>You can inject toolbars dynamically on click of a button.   |
| 3       | <b><u>External toolbars</u></b><br>The header and footer will be placed outside the page. These toolbars are positioned before and after the page inside the body, and they will remain in dom unless removed manually. |
| 4       | <b><u>Fixed toolbars</u></b><br>To set the toolbars position fixed, add the <i>data-position="fixed"</i> attribute to the header or footer.   |
| 5       | <b><u>Fullscreen position</u></b><br>To set the toolbars to fullscreen fixed position which overlays it over the content, add <i>data-fullscreen="true"</i> attribute to a fixed header.                                |
| 6       | <b><u>Fixed external toolbars</u></b><br>The header and footer can be made external by placing them outside the page and the toolbars position can also be made as a fixed position.                                    |
| 7       | <b><u>Persistent toolbars</u></b><br>These pages are an example of a persistent navbar in a fixed footer toolbar. You will see the page content transition while the header and footer on these pages remain in place.  |
| 8       | <b><u>Ajax optimized toolbars</u></b><br>The Ajax optimized persistant toolbars pages are optimized on the server side to check whether the request is from an Ajax.  |
| 9       | <b><u>Form in toolbar</u></b><br>The <i>form</i> can be used in toolbars, it can be placed in headers and footers by using your normal <i>&lt;form&gt;</i> .  |

## Toolbar

---

### Description

To enhance both headers and footers, you can use *toolbar* widget.

Following table lists different types of toolbar used in jQuery Mobile.

| Sr. No. | Toolbar Types & Description   |
|---------|---|
| 1       | <b><u>Markup</u></b><br>It contains toolbars like <i>header</i> that appears at the top and <i>footer</i> that appears at bottom of the page.   |
| 2       | <b><u>Theme</u></b><br>The theme swatch will be inherited by the header and footer toolbar from the page by default; however you can set the color of the theme swatch easily.  |
| 3       | <b><u>Buttons in toolbars</u></b><br>Buttons can be positioned on toolbars using <i>ui-btn-left</i> and <i>ui-btn-right</i> classes.  |
| 4       | <b><u>Adding back button to header</u></b><br>jQuery mobile framework generates a <i>back</i> button on a header when the header has a <i>data-add-back-btn="true"</i> attribute.                                       |
| 5       | <b><u>Navbars</u></b><br>You can use navigation bars inside the header and footer toolbars.   |
| 6       | <b><u>Grouped buttons</u></b><br>You can use <i>data-role="controlgroup"</i> and <i>data-type="horizontal"</i> attributes to group buttons into a set of button and wrap the links within an element.                   |
| 7       | <b><u>Adding padding</u></b><br>The toolbars will not have any padding for navbars and other widgets. To add padding, you can wrap content of the toolbars inside an element and add <i>uibar</i> class to the element. |

## Toolbar Markup

---

### Description

The toolbar markup contains toolbars like header that appears at the top and footer that appears at the bottom of the page. The *ui-title* class will be added to the headings by framework.

To maintain visual consistency, every heading with *ui-title* class will be identically styled by default. Both the header and footer is created using the *data-role* attribute.

## Example

Following example demonstrates the use of toolbar markup in jQuery Mobile.

```
<!DOCTYPE html>
<html>
<head>
    <title>Toolbar Markup</title>
    <meta name="viewport" content="width=device-width, initial-scale=1">
    <link rel="stylesheet"
    href="https://code.jquery.com/mobile/1.4.5/jquery.mobile-1.4.5.min.css">
    <script src="https://code.jquery.com/jquery-1.11.3.min.js"></script>
    <script src="https://code.jquery.com/mobile/1.4.5/jquery.mobile-
1.4.5.min.js"></script>
</head>
<body>
    <div data-role="header">
        <h2>Header</h2>
    </div>
    <h2>This is content of the page.</h2>
    <p>Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod
tempor incididunt ut labore et dolore magna aliqua. Ut enim ad minim veniam,
quis nostrud exercitation ullamco laboris nisi ut aliquip ex ea commodo
consequat</p>
    <div data-role="footer">
        <h2>Footer</h2>
    </div>
</body>
</html>
```

## Output

Let's carry out the following steps to see how the above code works:

- Save the above html code as **toolbar\_markup.html** file in your server root folder.
- Open this HTML file as [http://localhost/toolbar\\_markup.html](http://localhost/toolbar_markup.html) and the following output will be displayed.

## Header

### This is content of the page.

*Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua. Ut enim ad minim veniam, quis nostrud exercitation ullamco laboris nisi ut aliquip ex ea commodo consequat*

## Footer

## Toolbar Theme

### Description

The theme swatch will be inherited by the header and footer toolbar from the page by default; however, you can set the color of the theme swatch easily.

Whenever you use external fixed toolbars, you must set a theme always, as there will not be any parent page from which the theme can be inherited.

### Example

Following example demonstrates the use of toolbar theme in jQuery Mobile.

```
<!DOCTYPE html>
<html>
<head>
    <title>Toolbar Theme</title>
    <meta name="viewport" content="width=device-width, initial-scale=1">
    <link rel="stylesheet"
    href="https://code.jquery.com/mobile/1.4.5/jquery.mobile-1.4.5.min.css">
    <script src="https://code.jquery.com/jquery-1.11.3.min.js"></script>
    <script src="https://code.jquery.com/mobile/1.4.5/jquery.mobile-
    1.4.5.min.js"></script>
</head>
<body>
    <div data-role="page" id="page1">
        <div data-role="header" data-theme="b">
            <h2>Header</h2>
        </div>
```

```

<div data-role="main" class="ui-content">
    <h2>Page content</h2>
</div>
<div data-role="footer" data-theme="b">
    <h2>Footer</h2>
</div>
</div>
</body>
</html>

```

## Output

Let's carry out the following steps to see how the above code works:

- Save the above html code as **toolbar\_theme.html** file in your server root folder.
- Open this HTML file as [http://localhost/toolbar\\_theme.html](http://localhost/toolbar_theme.html) and the following output will be displayed.

**Header**

**Page Content**

**Footer**

## Buttons in Toolbars

---

### Description

Links in the toolbars will be automatically enhanced by the framework as buttons with mini and inline style; however, it has been deprecated in version 1.4.

The same applies for placing the buttons in header, if they are immediate header's child.

To place your buttons left or right in the header, you can use ui-btn-left and ui-btn-right respectively. The heading has some margin which provides the header bar its height. If you do not require the heading for your page, then add an element with class="ui-title".

### Example

Following example demonstrates the use of buttons in toolbars in jQuery Mobile.

```
<!DOCTYPE html>
<html>
<head>
    <title>Buttons in toolbars</title>
    <meta name="viewport" content="width=device-width, initial-scale=1">
    <link rel="stylesheet"
    href="https://code.jquery.com/mobile/1.4.5/jquery.mobile-1.4.5.min.css">
    <script src="https://code.jquery.com/jquery-1.11.3.min.js"></script>
    <script src="https://code.jquery.com/mobile/1.4.5/jquery.mobile-
1.4.5.min.js"></script>
</head>
<body>
    <div data-role="header">
        <a href="#" class="ui-btn-left ui-btn ui-btn-inline ui-mini ui-corner-all
ui-btn-icon-right ui-icon-gear">Options</a>
        <button class="ui-btn-right ui-btn ui-btn-b ui-btn-inline ui-mini ui-
corner-all ui-btn-icon-right ui-icon-check">Save</button>
        <span class="ui-title"></span>
    </div>
</body>
</html>
```

## Output

Let's carry out the following steps to see how the above code works:

- Save the above html code as **toolbar\_buttons.html** file in your server root folder.
- Open this HTML file as [http://localhost/toolbar\\_buttons.html](http://localhost/toolbar_buttons.html) and the following output will be displayed.



Options Save

Buttons can also be used in footers. For more information [click here](#).

## Buttons in Footers

---

### Description

The *ui-btn-left* and *ui-btn-right* classes cannot be used in the footer. However, you can still create the same appearance using custom CSS.

### Example

Following example demonstrates the use of buttons in footers in jQuery Mobile.

```
<!DOCTYPE html>
<html>
<head>
    <title>Buttons in footers</title>
    <meta name="viewport" content="width=device-width, initial-scale=1">
    <link rel="stylesheet"
    href="https://code.jquery.com/mobile/1.4.5/jquery.mobile-1.4.5.min.css">
    <script src="https://code.jquery.com/jquery-1.11.3.min.js"></script>
    <script src="https://code.jquery.com/mobile/1.4.5/jquery.mobile-
1.4.5.min.js"></script>
    <style>
        .footer-button-left,
        .footer-button-right {
            position: absolute;
            margin: 0;
            top: auto;
            bottom: 0.24em;
        }
        .footer-button-left {
            left: 0.4em;
        }
        .footer-button-right {
            right: 0.4em;
        }
    </style>
</head>
<body>
    <div data-role="footer">
```

```

<h2>Footer</h2>
<a href="#" class="ui-btn ui-corner-all ui-btn-inline ui-mini footer-
button-left ui-btn-icon-left ui-icon-home">Home</a>
<a href="#" class="ui-btn ui-corner-all ui-btn-inline ui-mini footer-
button-right ui-btn-icon-right ui-icon-gear">Options</a>
</div>
</body>
</html>

```

## Output

Let's carry out the following steps to see how the above code works:

- Save the above html code as **toolbar\_buttons\_footer.html** file in your server root folder.
- Open this HTML file as [http://localhost/toolbar\\_buttons\\_footer.html](http://localhost/toolbar_buttons_footer.html) and the following output will be displayed.

## Footer

[Home](#)

[Options](#)

## Adding Back Button to Header

---

### Description

jQuery mobile framework generates a back button on a header, when a header has a **data-add-back-btn="true"** attribute or the page plugin's addBackBtn option is set to true.

Adding **data-rel="back"** on an anchor will take you to the history entry, ignoring the default href.

You can provide the back button text using **data-back-btn-text="previous"** attribute or you can set plugin's options in the following way.

```
$.mobile.toolbar.prototype.options.backBtnText = "previous";
```

### Example

Following example demonstrates the use of adding a back button to the header in jQuery Mobile.

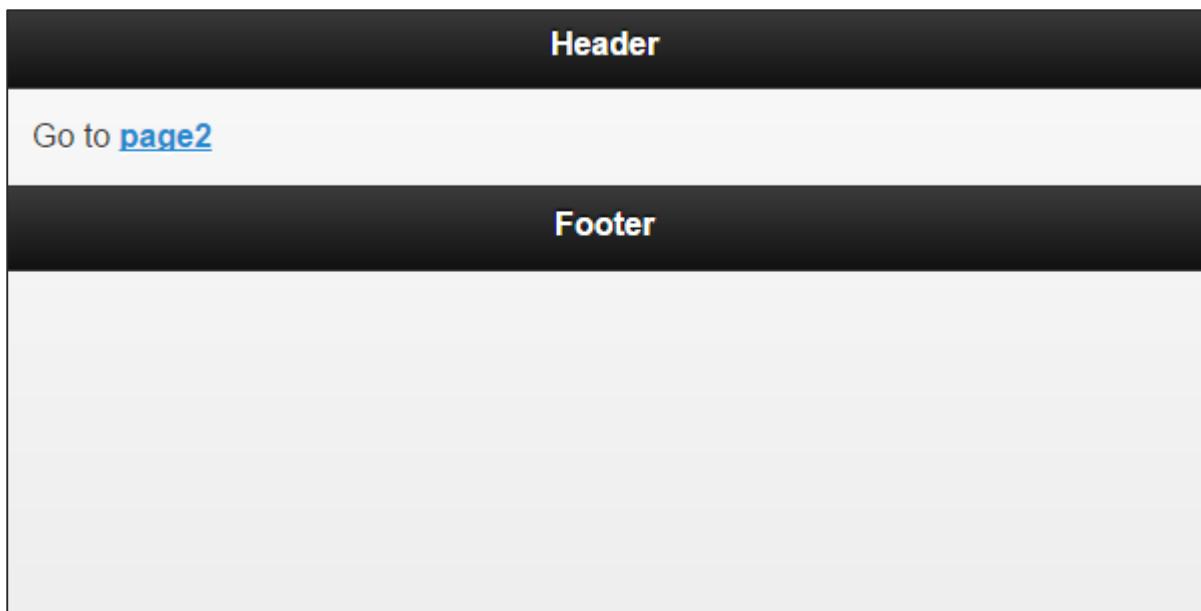
```
<!DOCTYPE HTML>
<html>
<head>
    <title>Adding back button to header</title>
    <link href="//code.jquery.com/mobile/1.2.0/jquery.mobile-1.2.0.min.css" rel="stylesheet">
    <script src="//code.jquery.com/jquery-1.8.1.min.js"></script>
    <script src="//code.jquery.com/mobile/1.2.0/jquery.mobile-1.2.0.min.js"></script>
</head>
<body>
    <div id="page1" data-role="page" data-add-back-btn="true">
        <div data-role="header">
            <h2>Header</h2>
        </div>
        <div data-role="content">
            Go to <a href="#page2">page2</a>
        </div>
        <div data-role="footer">
            <h2>Footer</h2>
        </div>
    </div>
    <div data-role="page" data-add-back-btn="true" data-back-btn-text="Previous" id="page2">
        <div data-role="header">
            <h2>Header</h2>
        </div>
        <div data-role="content">
            Go to <a href="#page1">page1</a>
        </div>
        <div data-role="footer">
            <h2>Footer</h2>
        </div>
    </div>
</body>
```

```
</html>
```

## Output

Let's carry out the following steps to see how the above code works:

- Save the above html code as **toolbar\_back.html** file in your server root folder.
- Open this HTML file as [http://localhost/toolbar\\_back.html](http://localhost/toolbar_back.html) and the following output will be displayed.



## Navbar Widget

---

### Description

The navbar widget is a set of buttons which links you to other web pages or sections. jQuery mobile provides a basic navbar widget consisting buttons up to 5 and an optional icon in a bar.

Following table lists navbar types used in jQuery mobile.

| Sr. No. | Navbar Types & Description  |
|---------|---|
| 1       | <u><b>Navbar basics</b></u><br>Navbar can be created using the <i>data-role="navbar"</i> attribute. |
| 2       | <u><b>Multi-row</b></u>   |

|    |  |
|----|--|
|    | Whenever a navbar consists of more than 5 items, all items will be wrapped to multiple lines two across.   |
| 3  | <p><b><u>Navbars in headers</u></b></p> <p>You can add navbar to the top of the page by adding the navbar container within the header block.</p>                                     |
| 4  | <p><b><u>Navbars in footers</u></b></p> <p>To add the navbar to the page's bottom, wrap the navbar within a container with a <i>data-role="footer"</i> attribute.</p>                |
| 5  | <p><b><u>Persistent</u></b></p> <p>The persistent navbar works like a tab bar that appears to be fixed as you browse across the pages.</p>   |
| 6  | <p><b><u>Icons</u></b></p> <p>To add icons to your navbar use <i>data-icon</i> attribute to each anchor.</p>   |
| 7  | <p><b><u>Icon position</u></b></p> <p>The navbar icons can be positioned on the navbar container instead of every single link.</p>   |
| 8  | <p><b><u>3rd party icon sets</u></b></p> <p>You can add third party icons to your navbar. To achieve this, just add custom styles to link to the icons and place them in navbar.</p> |
| 9  | <p><b><u>Themes</u></b></p> <p>Just like buttons, the theme swatch will be inherited by the navbars from their parent container.</p>   |
| 10 | <p><b><u>Navbars with button elements</u></b></p> <p>Instead of using links inside the navbar, you can also use button elements.</p>   |

## Navbar Basics

---

### Description

Navbar can be created using the ***data-role="navbar"*** attribute. When a navbar button is clicked it obtains the active state.

The ui-btn-active class will be removed from all navbar anchors and is added to the link which is activated. If this is a link to other page then, after the transition completion class will be removed again.

Depending upon the navbar items, the width of the browser will be evenly divided among the items.

## Example

Following example demonstrates the use of basic navbar in jQuery Mobile.

```
<!DOCTYPE html>
<html>
<head>
    <meta name="viewport" content="width=device-width, initial-scale=1">
    <link rel="stylesheet"
    href="https://code.jquery.com/mobile/1.4.5/jquery.mobile-1.4.5.min.css">
    <script src="https://code.jquery.com/jquery-1.11.3.min.js"></script>
    <script src="https://code.jquery.com/mobile/1.4.5/jquery.mobile-
1.4.5.min.js"></script>
</head>
<body>
    <p>Navbar will be rendered 100% with one item.</p>
    <div data-role="navbar">
        <ul>
            <li><a href="#" class="ui-btn-active">One</a></li>
        </ul>
    </div>
    <p>Navbar will be rendered 50% with two items.</p>
    <div data-role="navbar">
        <ul>
            <li><a href="#" class="ui-btn-active">One</a></li>
            <li><a href="#">Two</a></li>
        </ul>
    </div>
    <p>Each button will take 1/3 width of the browser window when navbar has 3
items.</p>
    <div data-role="navbar">
        <ul>
            <li><a href="#" class="ui-btn-active">One</a></li>
            <li><a href="#">Two</a></li>
            <li><a href="#">Three</a></li>
        </ul>
    </div>
    <p>Each button will take 1/4 width of the browser window when navbar has 4
items.</p>
```

```

<div data-role="navbar">
    <ul>
        <li><a href="#" class="ui-btn-active">One</a></li>
        <li><a href="#">Two</a></li>
        <li><a href="#">Three</a></li>
        <li><a href="#">Four</a></li>
    </ul>
</div>

<p>Each button will take 1/5 width of the browser window when navbar has 5 items.</p>
<div data-role="navbar">
    <ul>
        <li><a href="#" class="ui-btn-active">One</a></li>
        <li><a href="#">Two</a></li>
        <li><a href="#">Three</a></li>
        <li><a href="#">Four</a></li>
        <li><a href="#">Five</a></li>
    </ul>
</div>
</body>
</html>

```

## Output

Let's carry out the following steps to see how the above code works:

- Save the above html code as **navbar\_basic.html** file in your server root folder.
- Open this HTML file as [http://localhost/navbar\\_basic.html](http://localhost/navbar_basic.html) and the following output will be displayed.

Navbar will be rendered 100% with one item.

- [One](#)

Navbar will be rendered 50% with two items.

- [One](#)
- [Two](#)

Each button will take 1/3 width of the browser window when navbar has 3 items.

- [One](#)
- [Two](#)
- [Three](#)

Each button will take 1/4 width of the browser window when navbar has 4 items.

- [One](#)
- [Two](#)
- [Three](#)
- [Four](#)

Each button will take 1/5 width of the browser window when navbar has 5 items.

- [One](#)
- [Two](#)
- [Three](#)
- [Four](#)
- [Five](#)

## Navbar Multi-row

---

### Description

Whenever a navbar consists of more than 5 items, all items will be wrapped to multiple lines two across.

### Example

Following example demonstrates the use of multi-row navbar in jQuery Mobile.

```
<!DOCTYPE html>
<html>
<head>
<title>Multi-row navbar</title>
```

```

<meta name="viewport" content="width=device-width, initial-scale=1">
<link rel="stylesheet"
href="https://code.jquery.com/mobile/1.4.5/jquery.mobile-1.4.5.min.css">
<script src="https://code.jquery.com/jquery-1.11.3.min.js"></script>
<script src="https://code.jquery.com/mobile/1.4.5/jquery.mobile-
1.4.5.min.js"></script>
</head>
<body>
<div data-role="navbar">
<ul>
<li><a href="#" class="ui-btn-active">One</a></li>
<li><a href="#">Two</a></li>
<li><a href="#">Three</a></li>
<li><a href="#">Four</a></li>
<li><a href="#">Five</a></li>
<li><a href="#">Six</a></li>
<li><a href="#">Seven</a></li>
<li><a href="#">Eight</a></li>
</ul>
</div>
</body>
</html>

```

## Output

Let's carry out the following steps to see how the above code works:

- Save the above html code as **navbar\_multi\_row.html** file in your server root folder.
- Open this HTML file as [http://localhost/navbar\\_multi\\_row.html](http://localhost/navbar_multi_row.html) and the following output will be displayed.

- [One](#)
- [Two](#)
- [Three](#)
- [Four](#)
- [Five](#)
- [Six](#)
- [Seven](#)
- [Eight](#)

## Navbars in Headers

---

### Description

You can add the navbar to the top of the page by adding the navbar container within the header block.

### Example

Following example demonstrates the use of navbars in headers in jQuery Mobile.

```
<!DOCTYPE html>
<html>
<head>
    <title>Navbars in headers</title>
    <meta name="viewport" content="width=device-width, initial-scale=1">
    <link rel="stylesheet"
    href="https://code.jquery.com/mobile/1.4.5/jquery.mobile-1.4.5.min.css">
    <script src="https://code.jquery.com/jquery-1.11.3.min.js"></script>
    <script src="https://code.jquery.com/mobile/1.4.5/jquery.mobile-
1.4.5.min.js"></script>
</head>
<body>
    <div data-role="header" style="overflow:hidden;">
        <h2>This is Header</h2>
        <a href="#" data-icon="gear" class="ui-btn-right">Options</a>
        <div data-role="navbar">
            <ul>
                <li><a href="#" class="ui-btn-active">Home</a></li>
                <li><a href="#">About Us</a></li>
                <li><a href="#">Contact Us</a></li>
            </ul>
        </div>
    </div>
    <div data-role="content">
        Content goes here
    </div>
</body>
</html>
```

```

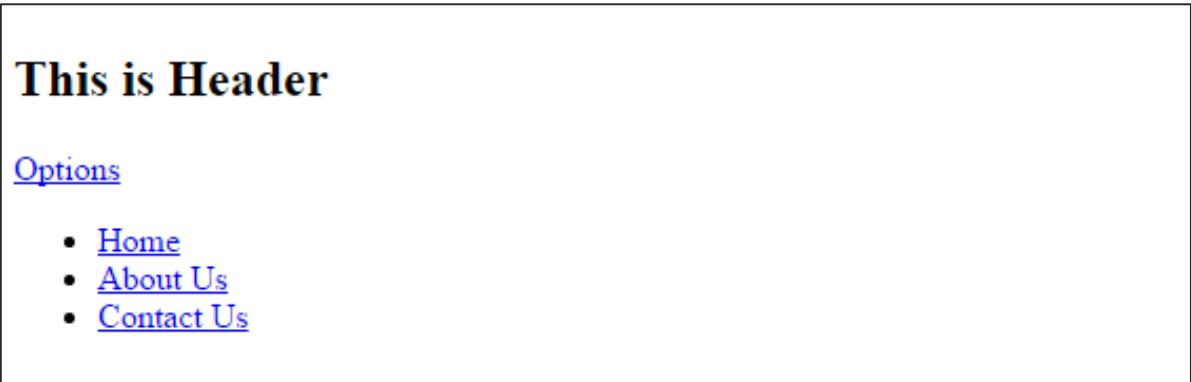
    </ul>
</div>
</div>
</body>
</html>

```

## Output

Let's carry out the following steps to see how the above code works:

- Save the above html code as **navbar\_headers.html** file in your server root folder.
- Open this HTML file as [http://localhost/navbar\\_headers.html](http://localhost/navbar_headers.html) and the following output will be displayed.



## This is Header

### Options

- [Home](#)
- [About Us](#)
- [Contact Us](#)

---

## Navbars in Footers

### Description

To add the navbar to the page's bottom, wrap the navbar within a container with a **data-role="footer"** attribute.

### Example

Following example demonstrates the use of **navbar in footer** in jQuery Mobile.

```

<!DOCTYPE html>
<html>
<head>
    <title>Navbars in footers</title>
    <meta name="viewport" content="width=device-width, initial-scale=1">
    <link rel="stylesheet"
    href="https://code.jquery.com/mobile/1.4.5/jquery.mobile-1.4.5.min.css">
    <script src="https://code.jquery.com/jquery-1.11.3.min.js"></script>

```

```

<script src="https://code.jquery.com/mobile/1.4.5/jquery.mobile-
1.4.5.min.js"></script>

</head>

<body>

<div data-role="footer" style="overflow:hidden;">
    <h2>This is Footer</h2>
    <div data-role="navbar">
        <ul>
            <li><a href="#" class="ui-btn-active">Home</a></li>
            <li><a href="#">About Us</a></li>
            <li><a href="#">Contact Us</a></li>
        </ul>
    </div>
</div>
</body>
</html>

```

## Output

Let's carry out the following steps to see how the above code works:

- Save the above html code as **navbar\_footers.html** file in your server root folder.
- Open this HTML file as [http://localhost/navbar\\_footers.html](http://localhost/navbar_footers.html) and the following output will be displayed.



## This is Footer

- [Home](#)
- [About Us](#)
- [Contact Us](#)

---

## Navbars Persistent

### Description

The persistent navbar works like a tab bar that appears to be fixed as you browse across the pages.

Adding class ui-btn-active to the anchor in your markup, you can set an active state for an item upon the navbar initialization.

Whenever the navbar is clicked, the content of the page changes and the persistent header and footer toolbar remains fixed. These toolbars must be placed on each page.

## Example

Following example demonstrates the use of navbar persistent in jQuery Mobile.

```
<!DOCTYPE html>
<html>
<head>
    <meta name="viewport" content="width=device-width, initial-scale=1">
    <link rel="stylesheet"
    href="https://code.jquery.com/mobile/1.4.5/jquery.mobile-1.4.5.min.css">
    <script src="https://code.jquery.com/jquery-1.11.3.min.js"></script>
    <script src="https://code.jquery.com/mobile/1.4.5/jquery.mobile-
1.4.5.min.js"></script>
<script>
$(function() {
    $( "[data-role='navbar']" ).navbar();
    $( "[data-role='header'], [data-role='footer']" ).toolbar();
});

// Update the the toolbar's contents
$( document ).on( "pagecontainerchange", function() {
    // Each of the 4 pages in this example has a data-title attribute
    // which value is equal to the nav button's text
    // For e.g., on first page: <div data-role="page" data-
title="Details">
    var current = $( ".ui-page-active" ).jqmData( "title" );
    // Change the heading
    $( "[data-role='header'] h1" ).text( current );
    // Remove active class
    $( "[data-role='navbar'] a.ui-btn-active" ).removeClass( "ui-btn-
active" );
    // Add active class to current nav button
    $( "[data-role='navbar'] a" ).each(function() {
        if ( $( this ).text() === current ) {
            $( this ).addClass( "ui-btn-active" );
        }
    })
})
```

```

    });
});

</script>
</head>
<body>

<div data-role="header" data-position="fixed" data-theme="a">
    <a href="" data-rel="back" class="ui-btn ui-btn-left ui-alt-icon ui-nodisc-icon ui-corner-all ui-btn-icon-notext ui-icon-carat-1">Back</a>
    <h1>Details</h1>
</div>
<div data-role="page" data-title="Details" class="jqm-demos">
    <div role="main" class="ui-content jqm-content jqm-fullwidth">
        <h1>jQuery Mobile</h1>

        <p>The jQuery Mobile is a user interface framework which is built on jQuery core and used for developing responsive websites or applications that are accessible on mobile, tablet and desktop devices. It uses features of both jQuery and jQueryUI to provide API features for mobile web applications.</p>
        <ul class="list">
            <li><p>The jQuery Mobile creates web applications in such a way that it will work the same way on the mobile, tablet and desktop devices.</p></li>
            <li><p>The jQuery Mobile is compatible with other frameworks such as <i>PhoneGap</i>, <i>Whitelight</i> etc.</p></li>
            <li><p>The jQuery Mobile provides set of touch friendly form inputs and UI widgets.</p></li>
            <li><p>The progressive enhancement brings you functionality to all mobile, tablet and desktop platforms and adds efficient page loads and wider device support.</p></li>
        </ul>
    </div>
</div>
<div data-role="footer" data-position="fixed" data-theme="a">
    <div data-role="navbar">
        <ul>
            <li><a href="/jquery_mobile/src/navbar_persistent.html" data-prefetch="true" data-transition="flip">Details</a></li>
            <li><a href="/jquery_mobile/src/page_one.html" data-prefetch="true" data-transition="fade">Friends</a></li>
            <li><a href="/jquery_mobile/src/page_two.html" data-prefetch="true" data-transition="slide">Messages</a></li>
            <li><a href="/jquery_mobile/src/page_three.html" data-prefetch="true" data-transition="turn">Emails</a></li>
        </ul>
    </div>
</div>

```

```

    </ul>
</div>
</div>
</body>
</html>

```

## Output

Let's carry out the following steps to see how the above code works:

- Save the above html code as **navbar\_persistent.html** file in your server root folder.
- Open this HTML file as [http://localhost/navbar\\_persistent.html](http://localhost/navbar_persistent.html) and the following output will be displayed.

[Back](#)

## Details

# jQuery Mobile

The jQuery Mobile is a user interface framework which is built on jQuery core and used for developing responsive websites or applications that are accessible on mobile, tablet and desktop devices. It uses features of both jQuery and jQueryUI to provide API features for mobile web applications.

- The jQuery Mobile creates web applications in such a way that it will work the same way on the mobile, tablet and desktop devices.
- The jQuery Mobile is compatible with other frameworks such as *PhoneGap*, *Whitelight* etc.
- The jQuery Mobile provides set of touch friendly form inputs and UI widgets.
- The progressive enhancement brings you functionality to all mobile, tablet and desktop platforms and adds efficient page loads and wider device support.
- [Details](#)
- [Friends](#)
- [Messages](#)
- [Emails](#)

## Navbars Icons

---

### Description

To add icons to your navbar use ***data-icon*** attribute to each anchor. By default, icons are placed at the top of the text.

### Example

Following example demonstrates the use of navbar icons in jQuery Mobile.

```
<!DOCTYPE html>
<head>
    <title>Navbar Icons</title>
    <meta name="viewport" content="width=device-width, initial-scale=1">
    <link rel="stylesheet"
    href="https://code.jquery.com/mobile/1.4.5/jquery.mobile-1.4.5.min.css">
    <script src="https://code.jquery.com/jquery-1.11.3.min.js"></script>
    <script src="https://code.jquery.com/mobile/1.4.5/jquery.mobile-
1.4.5.min.js"></script>
</head>
<body>
    <div data-role="navbar">
        <ul>
            <li><a href="#" data-icon="mail">Messages</a></li>
            <li><a href="#" data-icon="user" class="ui-btn-
active">Friends</a></li>
            <li><a href="#" data-icon="star">Notifications</a></li>
        </ul>
    </div>
</body>
</html>
```

### Output

Let's carry out the following steps to see how the above code works:

- Save the above html code as **navbar\_icons.html** file in your server root folder.
- Open this HTML file as [http://localhost/navbar\\_icons.html](http://localhost/navbar_icons.html) and the following output will be displayed.

- [Messages](#)
- [Friends](#)
- [Notifications](#)

## Navbars Icons Position

---

### Description

The navbar icons can be positioned on the navbar container instead of every single link. Navbar icons can be positioned left, right, top or bottom of the text using ***data-iconpos*** attribute.

### Example

Following example demonstrates the use of navbar icons position in jQuery Mobile.

```
<!DOCTYPE html>
<head>
    <title>Navbar Icons Position</title>
    <meta name="viewport" content="width=device-width, initial-scale=1">
    <link rel="stylesheet"
    href="https://code.jquery.com/mobile/1.4.5/jquery.mobile-1.4.5.min.css">
    <script src="https://code.jquery.com/jquery-1.11.3.min.js"></script>
    <script src="https://code.jquery.com/mobile/1.4.5/jquery.mobile-
    1.4.5.min.js"></script>
</head>
<body>
    <p>Icons placed at bottom:</p>
    <div data-role="navbar" data-iconpos="bottom">
        <ul>
            <li><a href="#" data-icon="mail">Messages</a></li>
            <li><a href="#" data-icon="user" class="ui-btn-
            active">Friends</a></li>
```

```

<li><a href="#" data-icon="star">Notifications</a></li>
</ul>
</div>
<p>Icons placed at top:</p>
<div data-role="navbar" data-iconpos="top">
<ul>
<li><a href="#" data-icon="mail">Messages</a></li>
<li><a href="#" data-icon="user" class="ui-btn-active">Friends</a></li>
<li><a href="#" data-icon="star">Notifications</a></li>
</ul>
</div>
<p>Icons placed at left:</p>
<div data-role="navbar" data-iconpos="left">
<ul>
<li><a href="#" data-icon="mail">Messages</a></li>
<li><a href="#" data-icon="user" class="ui-btn-active">Friends</a></li>
<li><a href="#" data-icon="star">Notifications</a></li>
</ul>
</div>
<p>Icons placed at right:</p>
<div data-role="navbar" data-iconpos="right">
<ul>
<li><a href="#" data-icon="mail">Messages</a></li>
<li><a href="#" data-icon="user" class="ui-btn-active">Friends</a></li>
<li><a href="#" data-icon="star">Notifications</a></li>
</ul>
</div>
</body>
</html>

```

## Output

Let's carry out the following steps to see how the above code works:

- Save the above html code as **navbar\_icons\_position.html** file in your server root folder.
- Open this HTML file as [http://localhost/navbar\\_icons\\_position.html](http://localhost/navbar_icons_position.html) and the following output will be displayed.

Icons placed at bottom:

- [Messages](#)
- [Friends](#)
- [Notifications](#)

Icons placed at top:

- [Messages](#)
- [Friends](#)
- [Notifications](#)

Icons placed at left:

- [Messages](#)
- [Friends](#)
- [Notifications](#)

Icons placed at right:

- [Messages](#)
- [Friends](#)
- [Notifications](#)

## Navbars 3rd Party Icon

---

### Description

You can add third party icons to your navbar. To achieve this, just add custom styles to link to the icons and place them in the navbar.

### Example

Following example demonstrates the use of navbars 3rd party icon in jQuery Mobile.

```
<!DOCTYPE html>
<html>
<head>
    <title>Navbars 3rd Party Icons</title>
    <meta name="viewport" content="width=device-width, initial-scale=1">
    <link rel="stylesheet"
        href="https://code.jquery.com/mobile/1.4.5/jquery.mobile-1.4.5.min.css">
```

```

<script src="https://code.jquery.com/jquery-1.11.3.min.js"></script>
<script src="https://code.jquery.com/mobile/1.4.5/jquery.mobile-
1.4.5.min.js"></script>
<style>
    .nav-glyphish-example .ui-btn { padding-top: 40px !important; }
    .nav-glyphish-example .ui-btn:after { width: 30px!important; height:
30px!important; margin-left: -15px !important; }
    #home:after { background: url("/jquery_mobile/images/home.png") no-repeat; }
    #notify:after { background:
url("/jquery_mobile/images/notifications.png") no-repeat; }
</style>
</head>
<body>
    <div data-role="footer" class="nav-glyphish-example" data-theme="a">
        <div data-role="navbar" class="nav-glyphish-example" data-grid="a">
            <ul>
                <li><a href="#" id="home" data-icon="custom">Home</a></li>
                <li><a href="#" id="notify" data-icon="custom">Notifications</a></li>

            </ul>
        </div>
    </div>
</body>
</html>

```

## Output

Let's carry out the following steps to see how the above code works:

- Save the above html code as **navbar\_3rd\_party\_icons.html** file in your server root folder.
- Open this HTML file as [http://localhost/navbar\\_3rd\\_party\\_icons.html](http://localhost/navbar_3rd_party_icons.html) and the following output will be displayed.

- [Home](#)
- [Notifications](#)

## Navbars Themes

---

### Description

Just like buttons, the theme swatch will be inherited by the navbars from their parent container.

Whenever the navbar is positioned in the header or footer toolbar, the default toolbar swatch a will be inherited unless it is set in the markup.

You can add swatch classes to apply the body swatch (ui-body-a and ui-body-b). You can also use data-theme attribute to set theme color for individual navbar items and specify a theme swatch.

The class ui-body adds the standard body padding.

### Example

Following example demonstrates the use of navbar themes in jQuery Mobile.

```
<!DOCTYPE html>
<html>
<head>
    <title>Navbars Themes</title>
    <meta name="viewport" content="width=device-width, initial-scale=1">
    <link rel="stylesheet"
    href="https://code.jquery.com/mobile/1.4.5/jquery.mobile-1.4.5.min.css">
    <script src="https://code.jquery.com/jquery-1.11.3.min.js"></script>
    <script src="https://code.jquery.com/mobile/1.4.5/jquery.mobile-
1.4.5.min.js"></script>
</head>
<body>
    <div class="ui-body-a ui-body">
        <h3>Swatch "a"</h3>
        <div data-role="navbar">
```

```

<ul>
    <li><a href="#" data-icon="star">A</a></li>
    <li><a href="#" data-icon="gear">B</a></li>
    <li><a href="#" data-icon="grid">C</a></li>
    <li><a href="#" data-icon="arrow-l">D</a></li>
    <li><a href="#" data-icon="arrow-r">E</a></li>
</ul>
</div>
</div>
<div class="ui-body-b ui-body">
    <h3>Swatch "b"</h3>
    <div data-role="navbar">
        <ul>
            <li><a href="#" data-icon="star">A</a></li>
            <li><a href="#" data-icon="gear">B</a></li>
            <li><a href="#" data-icon="grid">C</a></li>
            <li><a href="#" data-icon="arrow-l">D</a></li>
            <li><a href="#" data-icon="arrow-r">E</a></li>
        </ul>
    </div>
</div>
<h3><i>data-theme</i> attribute</h3>
<div data-role="footer">
    <div data-role="navbar">
        <ul>
            <li><a href="#" data-icon="grid" data-theme="a">A</a></li>
            <li><a href="#" data-icon="grid" data-theme="b">B</a></li>
        </ul>
    </div>
</div>
</body>
</html>

```

## Output

Let's carry out the following steps to see how the above code works:

- Save the above html code as **navbar\_themes.html** file in your server root folder.

- Open this HTML file as [http://localhost/navbar\\_themes.html](http://localhost/navbar_themes.html) and the following output will be displayed.

### Swatch "a"

- [A](#)
- [B](#)
- [C](#)
- [D](#)
- [E](#)

### Swatch "b"

- [A](#)
- [B](#)
- [C](#)
- [D](#)
- [E](#)

### *data-theme* attribute

- [A](#)
- [B](#)

## Navbars with Button Elements

---

### Description

Instead of using links inside the navbar, you can also use button elements.

### Example

Following example demonstrates the use of navbar with button in jQuery Mobile.

```
<!DOCTYPE html>
<html>
<head>
  <title>Navbars with button elements</title>
  <meta name="viewport" content="width=device-width, initial-scale=1">
  <link rel="stylesheet"
  href="https://code.jquery.com/mobile/1.4.5/jquery.mobile-1.4.5.min.css">
  <script src="https://code.jquery.com/jquery-1.11.3.min.js"></script>
```

```

<script src="https://code.jquery.com/mobile/1.4.5/jquery.mobile-
1.4.5.min.js"></script>

</head>

<body>

<div data-role="navbar" data-iconpos="left">
    <ul>
        <li><button data-icon="star">Button One</button></li>
        <li><button data-icon="arrow-l" data-theme="b">Button
Two</button></li>
        <li><button data-icon="arrow-r">Button Three</button></li>
    </ul>
</div>
</body>
</html>

```

## Output

Let's carry out the following steps to see how the above code works:

- Save the above html code as **navbar\_with\_button.html** file in your server root folder.
- Open this HTML file as [http://localhost/navbar\\_with\\_button.html](http://localhost/navbar_with_button.html) and the following output will be displayed.



## Grouped Buttons

---

### Description

You can use ***data-role="controlgroup"*** and ***data-type="horizontal"*** attributes to group buttons into a set of button and wrap the links within an element.

### Example

Following example demonstrates the use of grouped buttons in jQuery Mobile.

```
<!DOCTYPE html>
<html>
<head>
    <title>Grouped Buttons</title>
    <meta name="viewport" content="width=device-width, initial-scale=1">
    <link rel="stylesheet"
    href="https://code.jquery.com/mobile/1.4.5/jquery.mobile-1.4.5.min.css">
    <script src="https://code.jquery.com/jquery-1.11.3.min.js"></script>
    <script src="https://code.jquery.com/mobile/1.4.5/jquery.mobile-
1.4.5.min.js"></script>
</head>
<body>
    <div data-role="header">
        <div data-role="controlgroup" data-type="horizontal" class="ui-mini ui-
btn-left">
            <a href="#" class="ui-btn ui-btn-icon-right ui-icon-gear">Options</a>
            <a href="#" class="ui-btn ui-btn-icon-right ui-icon-plus">Add</a>
            <a href="#" class="ui-btn ui-btn-icon-right ui-icon-grid">grid</a>
        </div>
        <h2>Header</h2>
    </div>
</body>
</html>
```

## Output

Let's carry out the following steps to see how the above code works:

- Save the above html code as **toolbar\_grouped\_buttons.html** file in your server root folder.
- Open this HTML file as [http://localhost/toolbar\\_grouped\\_buttons.html](http://localhost/toolbar_grouped_buttons.html) and the following output will be displayed.

[Options Add grid](#)

## Header

## Adding a Padding

### Description

The toolbars will not have any padding for navbars and other widgets.

To add padding, you can wrap the content of the toolbars inside an element and add ui-bar class to the element. You can also use your own custom CSS for padding.

### Example

Following example demonstrates the use of adding padding in jQuery Mobile.

```
<!DOCTYPE html>
<html>
<head>
    <title>Adding padding</title>
    <meta name="viewport" content="width=device-width, initial-scale=1">
    <link rel="stylesheet" href="https://code.jquery.com/mobile/1.4.5/jquery.mobile-1.4.5.min.css">
    <script src="https://code.jquery.com/jquery-1.11.3.min.js"></script>
    <script src="https://code.jquery.com/mobile/1.4.5/jquery.mobile-1.4.5.min.js"></script>
</head>
<body>
    <div data-role="footer">
        <div class="ui-bar">
            <label for="select-choice-1" class="ui-hidden-accessible">City:</label>
            <select name="select-choice-1" id="select-choice" data-mini="true" data-inline="true">
                <option value="India">Bengaluru</option>

```

```

        <option value="Norway">Oslo</option>
        <option value="Mauritius">Port Louis</option>
    </select>
</div>
</div>
</body>
</html>
```

## Output

Let's carry out the following steps to see how the above code works:

- Save the above html code as **toolbar\_adding\_padding.html** file in your server root folder.
- Open this HTML file as [http://localhost/toolbar\\_adding\\_padding.html](http://localhost/toolbar_adding_padding.html) and the following output will be displayed.

City: **Bengaluru ▾**

## Dynamic Toolbars

---

### Description

You can inject toolbars dynamically on the click of a button. You need to update the page height and padding by invoking the following function:

```
$.mobile.resetActivePageHeight()
```

### Example

Following example demonstrates the use of dynamic toolbars in jQuery Mobile.

```
<!DOCTYPE html>
<html>
<head>
    <title>Dynamic Toolbars</title>
    <meta name="viewport" content="width=device-width, initial-scale=1">
    <link rel="stylesheet" href="https://code.jquery.com/mobile/1.4.5/jquery.mobile-1.4.5.min.css">
    <script src="https://code.jquery.com/jquery-1.11.3.min.js"></script>
    <script src="https://code.jquery.com/mobile/1.4.5/jquery.mobile-1.4.5.min.js"></script>
<script>
    $( document ).on( "click", "#inject-toolbars", function() {
        $( "<div data-role='header'><h2>Dynamic Header</h2></div>" )
            .prependTo( "#page-with-dynamic-toolbars" )
            .toolbar({ position: "fixed" });

        $( "<div data-role='footer'><h2>Dynamic Footer</h2></div>" )
            .appendTo( "#page-with-dynamic-toolbars" )
            .toolbar({ position: "fixed" });

        $.mobile.resetActivePageHeight();
    });
</script>
</head>
<body>
    <div data-role="page" class="jqm-demos" id="page-with-dynamic-toolbars" data-url="page-with-dynamic-toolbars">
        <div role="main" class="ui-content jqm-content jqm-fullwidth">
            <h2>Page content</h2>
            <p>Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua. Ut enim ad minim veniam, quis nostrud exercitation ullamco laboris nisi ut aliquip ex ea commodo consequat. Duis aute irure dolor in reprehenderit in voluptate velit esse cillum dolore eu fugiat nulla pariatur.</p>
            <div data-demo-html="#page-with-dynamic-toolbars" data-demo-js="true">
                <button id="inject-toolbars" class="ui-btn ui-btn-inline ui-corner-all">Inject toolbars</button>
            </div>
        </div>
    </div>
</body>
```

```
</body>
</html>
```

## Output

Let's carry out the following steps to see how the above code works:

- Save the above html code as **toolbar\_widget\_dynamic.html** file in your server root folder.
- Open this HTML file as [http://localhost/toolbar\\_widget\\_dynamic.html](http://localhost/toolbar_widget_dynamic.html) and the following output will be displayed.

## Page content

*Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua. Ut enim ad minim veniam, quis nostrud exercitation ullamco laboris nisi ut aliquip ex ea commodo consequat. Duis aute irure dolor in reprehenderit in voluptate velit esse cillum dolore eu fugiat nulla pariatur.*

[Inject toolbars](#)

## External Toolbars

### Description

The header and footer will be placed outside the page. These toolbars are positioned before and after the page inside the body and they will remain in dom unless removed manually.

Since the toolbars are outside the page, they will not be auto initialized. You have to call the toolbar plugin by invoking the following function.

```
$(function(){
  $('[data-role='header'], [data-role='footer']").toolbar();
});
```

As the toolbars are placed outside the page, they will not inherit a theme from the page, instead you have to set a theme for them always.

The **data-theme** attribute can be used for this or set theme options whenever you call the plugin by invoking the following function.

```
$(function(){
```

```
$( "[data-role='header']" ).toolbar({ theme: "a" });
});
```

## Example

Following example demonstrates the use of external toolbars in jQuery Mobile.

```
<!DOCTYPE html>
<html>
<head>
    <title>Toolbar External Toolbars</title>
    <meta name="viewport" content="width=device-width, initial-scale=1">
    <link rel="stylesheet"
    href="https://code.jquery.com/mobile/1.4.5/jquery.mobile-1.4.5.min.css">
    <script src="https://code.jquery.com/jquery-1.11.3.min.js"></script>
    <script src="https://code.jquery.com/mobile/1.4.5/jquery.mobile-
1.4.5.min.js"></script>
    <script>
        $(function(){
            $( "[data-role='header'], [data-role='footer']" ).toolbar({ theme: "a" });
        });
    </script>
</head>
<body>
    <div data-role="header">
        <h2>External Header</h2>
    </div>
    <div data-role="page">
        <h2>Page's Content</h2>
        <p>Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua. Ut enim ad minim veniam, quis nostrud exercitation ullamco laboris nisi ut aliquip ex ea commodo consequat. Duis aute irure dolor in reprehenderit in voluptate velit esse cillum dolore eu fugiat nulla pariatur. </p>
    </div>
    <div data-role="footer">
        <h2>External Footer</h2>
    </div>
</body>
</html>
```

## Output

Let's carry out the following steps to see how the above code works:

- Save the above html code as **toolbar\_external.html** file in your server root folder.
- Open this HTML file as [http://localhost/toolbar\\_external.html](http://localhost/toolbar_external.html) and the following output will be displayed.

### External Header

### Page's Content

*Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua. Ut enim ad minim veniam, quis nostrud exercitation ullamco laboris nisi ut aliquip ex ea commodo consequat. Duis aute irure dolor in reprehenderit in voluptate velit esse cillum dolore eu fugiat nulla pariatur.*

### External Footer

## Fixed Position Toolbars

---

### Description

To set the toolbars position fixed, add the ***data-position="fixed"*** attribute to the header or footer.

### Example

Following example demonstrates the use of fixed position toolbars in jQuery Mobile.

```
<!DOCTYPE html>
<html>
<head>
    <title>Fixed Position Toolbars</title>
    <meta name="viewport" content="width=device-width, initial-scale=1">
    <link rel="stylesheet"
    href="https://code.jquery.com/mobile/1.4.5/jquery.mobile-1.4.5.min.css">
    <script src="https://code.jquery.com/jquery-1.11.3.min.js"></script>
    <script src="https://code.jquery.com/mobile/1.4.5/jquery.mobile-
    1.4.5.min.js"></script>
</head>
```

```
<body>
  <div data-role="header" data-position="fixed">
    <h2>Fixed Header</h2>
  </div>
  <h2>Most widely spoken languages</h2>
  <ul>
    <li><p>Mandarin(Standard Chinese)</p></li>
    <li><p>Spanish</p></li>
    <li><p>English</p></li>
    <li><p>Hindi</p></li>
    <li><p>Arabic</p></li>
    <li><p>Portuguese</p></li>
    <li><p>Bengali</p></li>
    <li><p>Russian</p></li>
    <li><p>Japanese</p></li>
    <li><p>Punjabi</p></li>
    <li><p>German</p></li>
    <li><p>Javanese</p></li>
    <li><p>Wu (Shanghainese)</p></li>
    <li><p>Malay/Indonesian</p></li>
    <li><p>Telugu</p></li>
    <li><p>Vietnamese</p></li>
    <li><p>Korean</p></li>
    <li><p>French</p></li>
    <li><p>Marathi</p></li>
    <li><p>Tamil</p></li>
    <li><p>Urdu</p></li>
    <li><p>Turkish </p></li>
    <li><p>Italian</p></li>
    <li><p>Yue (Cantonese)</p></li>
    <li><p>Thai</p></li>
    <li><p>Gujarati</p></li>
    <li><p>Jin</p></li>
    <li><p>Southern</p></li>
    <li><p>Persian</p></li>
    <li><p>Polish</p></li>
    <li><p>Pashto</p></li>
```

```
<li><p>Kannada</p></li>
<li><p>Xiang (Hunnanese)</p></li>
<li><p>Malayalam</p></li>
<li><p>Sundanese</p></li>
<li><p>Hausa</p></li>
<li><p>Odia (Oriya)</p></li>
<li><p>Burmese</p></li>
<li><p>Hakka</p></li>
<li><p>Ukrainian</p></li>
</ul>
<div data-role="footer" data-position="fixed">
    <h2>Fixed Footer</h2>
</div>
</body>
</html>
```

## Output

Let's carry out the following steps to see how the above code works:

- Save the above html code as **toolbar\_fixed.html** file in your server root folder.
- Open this HTML file as [http://localhost/toolbar\\_fixed.html](http://localhost/toolbar_fixed.html) and the following output will be displayed.

## Fixed Header

### Most widely spoken languages

- Mandarin(Standard Chinese)
- Spanish
- English
- Hindi
- Arabic
- Portuguese
- Bengali
- Russian
- Japanese
- Punjabi
- German
- Javanese
- Wu (Shanghainese)

- Wu (Shanghainese)
- Malay/Indonesian
- Telugu
- Vietnamese
- Korean
- French
- Marathi
- Tamil
- Urdu
- Turkish
- Italian
- Yue (Cantonese)
- Thai
- Gujarati

- Jin
- Southern
- Persian
- Polish
- Pashto
- Kannada
- Xiang (Hunanese)
- Malayalam
- Sundanese
- Hausa
- Odia (Oriya)
- Burmese
- Hakka
- Ukrainian

## Fixed Footer

## Fullscreen Position Toolbars

---

### Description

To set the toolbars to fullscreen fixed position which overlays it over the content, add **`data-fullscreen="true"`** attribute to a fixed header.

### Example

Following example demonstrates the use of fullscreen position toolbars in jQuery Mobile.

```
<!DOCTYPE html>
<html>
<head>
  <title>Fullscreen Position Toolbars</title>
  <meta name="viewport" content="width=device-width, initial-scale=1">
  <link rel="stylesheet"
  href="https://code.jquery.com/mobile/1.4.5/jquery.mobile-1.4.5.min.css">
  <script src="https://code.jquery.com/jquery-1.11.3.min.js"></script>
  <script src="https://code.jquery.com/mobile/1.4.5/jquery.mobile-
  1.4.5.min.js"></script>
```

```
</head>
<body>
  <div data-role="header" data-position="fixed" data-fullscreen="true">
    <h2>Fixed Header</h2>
  </div>
  <h2>Most widely spoken languages</h2>
  <ul>
    <li><p>Mandarin (Standard Chinese)</p></li>
    <li><p>Spanish</p></li>
    <li><p>English</p></li>
    <li><p>Hindi</p></li>
    <li><p>Arabic</p></li>
    <li><p>Portuguese</p></li>
    <li><p>Bengali</p></li>
    <li><p>Russian</p></li>
    <li><p>Japanese</p></li>
    <li><p>Punjabi</p></li>
    <li><p>German</p></li>
    <li><p>Javanese</p></li>
    <li><p>Wu (Shanghainese)</p></li>
    <li><p>Malay/Indonesian</p></li>
    <li><p>Telugu</p></li>
    <li><p>Vietnamese</p></li>
    <li><p>Korean</p></li>
    <li><p>French</p></li>
    <li><p>Marathi</p></li>
    <li><p>Tamil</p></li>
    <li><p>Urdu</p></li>
    <li><p>Turkish </p></li>
    <li><p>Italian</p></li>
    <li><p>Yue (Cantonese)</p></li>
    <li><p>Thai</p></li>
    <li><p>Gujarati</p></li>
    <li><p>Jin</p></li>
    <li><p>Southern</p></li>
    <li><p>Persian</p></li>
    <li><p>Polish</p></li>
```

```
<li><p>Pashto</p></li>
<li><p>Kannada</p></li>
<li><p>Xiang (Hunnanese)</p></li>
<li><p>Malayalam</p></li>
<li><p>Sundanese</p></li>
<li><p>Hausa</p></li>
<li><p>Odia (Oriya)</p></li>
<li><p>Burmese</p></li>
<li><p>Hakka</p></li>
<li><p>Ukrainian</p></li>
</ul>
<div data-role="footer" data-position="fixed" data-fullscreen="true">
    <h2>Fixed Footer</h2>
</div>
</body>
</html>
```

## Output

Let's carry out the following steps to see how the above code works:

- Save the above html code as **toolbar\_fullscreen.html** file in your server root folder.
- Open this HTML file as <http://localhost/toolbarFullscreen.html> and the following output will be displayed.

## Fixed Header

### Most widely spoken languages

- Mandarin(Standard Chinese)
- Spanish
- English
- Hindi
- Arabic
- Portuguese
- Bengali
- Russian
- Japanese
- Punjabi
- German
- Javanese
- Wu (Shanghainese)
- Malay/Indonesian
- Telugu

- Vietnamese
- Korean
- French
- Marathi
- Tamil
- Urdu
- Turkish
- Italian
- Yue (Cantonese)
- Thai
- Gujarati
- Jin
- Southern

- Persian
- Polish
- Pashto
- Kannada
- Xiang (Hunanese)
- Malayalam
- Sundanese
- Hausa
- Odia (Oriya)
- Burmese
- Hakka
- Ukrainian

## Fixed Footer

## External Fixed Toolbars

---

### Description

The header and footer can be made external by placing them outside the page and the position of the toolbars position can also be fixed.

### Example

Following example demonstrates the use of external fixed toolbars in jQuery Mobile.

```
<!DOCTYPE html>
<html>
<head>
    <title>Toolbar External Toolbars</title>
    <meta name="viewport" content="width=device-width, initial-scale=1">
    <link rel="stylesheet"
    href="https://code.jquery.com/mobile/1.4.5/jquery.mobile-1.4.5.min.css">
    <script src="https://code.jquery.com/jquery-1.11.3.min.js"></script>
    <script src="https://code.jquery.com/mobile/1.4.5/jquery.mobile-
    1.4.5.min.js"></script>
    <script>
```

```
$($.function(){
    $(".[data-role='header'], [data-role='footer']").toolbar();
});
</script>
</head>
<body>
    <div data-role="header" data-position="fixed" data-theme="a">
        <h1>External fixed header</h1>
    </div>
    <div data-role="page">
        <h2>Most widely spoken languages</h2>
        <ul>
            <li><p>Mandarin(Standard Chinese)</p></li>
            <li><p>Spanish</p></li>
            <li><p>English</p></li>
            <li><p>Hindi</p></li>
            <li><p>Arabic</p></li>
            <li><p>Portuguese</p></li>
            <li><p>Bengali</p></li>
            <li><p>Russian</p></li>
            <li><p>Japanese</p></li>
            <li><p>Punjabi</p></li>
            <li><p>German</p></li>
            <li><p>Javanese</p></li>
            <li><p>Wu (Shanghainese)</p></li>
            <li><p>Malay/Indonesian</p></li>
            <li><p>Telugu</p></li>
            <li><p>Vietnamese</p></li>
            <li><p>Korean</p></li>
            <li><p>French</p></li>
            <li><p>Marathi</p></li>
            <li><p>Tamil</p></li>
            <li><p>Urdu</p></li>
            <li><p>Turkish</p></li>
            <li><p>Italian</p></li>
            <li><p>Yue (Cantonese)</p></li>
            <li><p>Thai</p></li>
        </ul>
    </div>
</body>
```

```

<li><p>Gujarati</p></li>
<li><p>Jin</p></li>
<li><p>Southern</p></li>
<li><p>Persian</p></li>
<li><p>Polish</p></li>
<li><p>Pashto</p></li>
<li><p>Kannada</p></li>
<li><p>Xiang (Hunnanese)</p></li>
<li><p>Malayalam</p></li>
<li><p>Sundanese</p></li>
<li><p>Hausa</p></li>
<li><p>Odia (Oriya)</p></li>
<li><p>Burmese</p></li>
<li><p>Hakka</p></li>
<li><p>Ukrainian</p></li>
</ul>
</div>
<div data-role="footer" data-position="fixed" data-theme="a">
    <h1>External fixed footer</h1>
</div>
</body>
</html>

```

## Output

Let's carry out the following steps to see how the above code works:

- Save the above html code as **toolbar\_external\_fixed.html** file in your server root folder.
- Open this HTML file as [http://localhost/toolbar\\_external\\_fixed.html](http://localhost/toolbar_external_fixed.html) and the following output will be displayed.

## External fixed header

### Most widely spoken languages

- Mandarin(Standard Chinese)
- Spanish
- English
- Hindi
- Arabic
- Portuguese
- Bengali
- Russian
- Japanese
- Punjabi
- German
- Javanese
- Wu (Shanghainese)

- Malay/Indonesian
- Telugu
- Vietnamese
- Korean
- French
- Marathi
- Tamil
- Urdu
- Turkish
- Italian
- Yue (Cantonese)
- Thai
- Gujarati
- Jin

- Southern
- Persian
- Polish
- Pashto
- Kannada
- Xiang (Hunanese)
- Malayalam
- Sundanese
- Hausa
- Odia (Oriya)
- Burmese
- Hakka
- Ukrainian

## External fixed footer

---

## Persistent Toolbars

---

### Description

These pages are example of a persistent navbar in a fixed footer toolbar. You will see the page content transition while the header and footer on these pages remain in place. For example, please refer [navbar persistent](#) chapter.

---

## Navbars Persistent

---

### Description

The persistent navbar works like a tab bar that appears to be fixed as you browse across the pages. Adding class **ui-btn-active** to the anchor in your markup, you can set an active state for an item upon the navbar initialization.

Whenever the navbar is clicked, the content of the page changes and the persistent header and footer toolbar remains fixed. These toolbars must be placed on each page.

## Example

Following example demonstrates the use of navbar persistent in jQuery Mobile.

```
<!DOCTYPE html>
<html>
<head>
    <meta name="viewport" content="width=device-width, initial-scale=1">
    <link rel="stylesheet"
    href="https://code.jquery.com/mobile/1.4.5/jquery.mobile-1.4.5.min.css">
    <script src="https://code.jquery.com/jquery-1.11.3.min.js"></script>
    <script src="https://code.jquery.com/mobile/1.4.5/jquery.mobile-
1.4.5.min.js"></script>
<script>
    $(function() {
        $( "[data-role='navbar']" ).navbar();
        $( "[data-role='header'], [data-role='footer']" ).toolbar();
    });
    // Update the toolbar's contents
    $( document ).on( "pagecontainerchange", function() {
        // Each of the 4 pages in this example has a data-title attribute
        // which value is equal to the nav button's text
        // For e.g., on first page: <div data-role="page" data-
title="Details">
        var current = $( ".ui-page-active" ).jqmData( "title" );
        // Change the heading
        $( "[data-role='header'] h1" ).text( current );
        // Remove active class
        $( "[data-role='navbar'] a.ui-btn-active" ).removeClass( "ui-btn-
active" );
        // Add active class to current nav button
        $( "[data-role='navbar'] a" ).each(function() {
            if ( $( this ).text() === current ) {
                $( this ).addClass( "ui-btn-active" );
            }
        });
    });
</script>
</head>
```

```

<body>
    <div data-role="header" data-position="fixed" data-theme="a">
        <a href="" data-rel="back" class="ui-btn ui-btn-left ui-alt-icon ui-nodisc-icon ui-corner-all ui-btn-icon-notext ui-icon-carat-1">Back</a>
        <h1>Details</h1>
    </div>
    <div data-role="page" data-title="Details" class="jqm-demos">
        <div role="main" class="ui-content jqm-content jqm-fullwidth">
            <h1>jQuery Mobile</h1>
            <p>The jQuery Mobile is a user interface framework which is built on jQuery core and used for developing responsive websites or applications that are accessible on mobile, tablet and desktop devices. It uses features of both jQuery and jQueryUI to provide API features for mobile web applications.</p>
            <ul class="list">
                <li><p>The jQuery Mobile creates web applications in such a way that it will work the same way on the mobile, tablet and desktop devices.</p></li>
                <li><p>The jQuery Mobile is compatible with other frameworks such as <i>PhoneGap</i>, <i>Whitelight</i> etc.</p></li>
                <li><p>The jQuery Mobile provides set of touch friendly form inputs and UI widgets.</p></li>
                <li><p>The progressive enhancement brings you functionality to all mobile, tablet and desktop platforms and adds efficient page loads and wider device support.</p></li>
            </ul>
        </div>
    </div>
    <div data-role="footer" data-position="fixed" data-theme="a">
        <div data-role="navbar">
            <ul>
                <li><a href="/jquery_mobile/src/navbar_persistent.html" data-prefetch="true" data-transition="flip">Details</a></li>
                <li><a href="/jquery_mobile/src/page_one.html" data-prefetch="true" data-transition="fade">Friends</a></li>
                <li><a href="/jquery_mobile/src/page_two.html" data-prefetch="true" data-transition="slide">Messages</a></li>
                <li><a href="/jquery_mobile/src/page_three.html" data-prefetch="true" data-transition="turn">Emails</a></li>
            </ul>
        </div>
    </div>
</body>

```

```
</html>
```

## Output

Let's carry out the following steps to see how the above code works:

- Save the above html code as **navbar\_persistent.html** file in your server root folder.
- Open this HTML file as [http://localhost/navbar\\_persistent.html](http://localhost/navbar_persistent.html) and the following output will be displayed.

[Back](#)

## Details

# jQuery Mobile

The jQuery Mobile is a user interface framework which is built on jQuery core and used for developing responsive websites or applications that are accessible on mobile, tablet and desktop devices. It uses features of both jQuery and jQueryUI to provide API features for mobile web applications.

- The jQuery Mobile creates web applications in such a way that it will work the same way on the mobile, tablet and desktop devices.
- The jQuery Mobile is compatible with other frameworks such as *PhoneGap*, *Whitelight* etc.
- The jQuery Mobile provides set of touch friendly form inputs and UI widgets.
- The progressive enhancement brings you functionality to all mobile, tablet and desktop platforms and adds efficient page loads and wider device support.
- [Details](#)
- [Friends](#)
- [Messages](#)
- [Emails](#)

## Ajax Optimized Toolbars

---

### Description

Ajax optimized persistant toolbars pages are optimized on the server side to check whether the request is from an Ajax.

If the request is from Ajax only, the actual page div is sent instead of the entire page.

When you navigate to any navbar page at the bottom, you can inspect the return data which contains no head, toolbars, html tag, or body tag.

However, on refreshing the page you will see all these things present.

This can be achieved by checking the HTTP\_X\_REQUESTED\_WITH header using the following if case:

```
if (!isset($_SERVER['HTTP_X_REQUESTED_WITH']) ||  
strtolower($_SERVER['HTTP_X_REQUESTED_WITH']) != 'xmlhttprequest') {
```

### Example

Following example demonstrates the use of ajax optimized toolbars in jQuery Mobile.

```
<?php if (!isset($_SERVER['HTTP_X_REQUESTED_WITH']) ||  
strtolower($_SERVER['HTTP_X_REQUESTED_WITH']) != 'xmlhttprequest') { ?>  
<!DOCTYPE html>  
<html>  
<head>  
    <title>Ajax optimized persistent toolbars</title>  
    <meta name="viewport" content="width=device-width, initial-scale=1">  
    <link rel="stylesheet"  
    href="https://code.jquery.com/mobile/1.4.5/jquery.mobile-1.4.5.min.css">  
    <script src="https://code.jquery.com/jquery-1.11.3.min.js"></script>  
    <script src="https://code.jquery.com/mobile/1.4.5/jquery.mobile-  
1.4.5.min.js"></script>  
    <script>  
        $(function(){  
            $('[data-role='navbar']").navbar();  
            $('[data-role='header'], [data-role='footer']").toolbar();  
        });  
    </script>  
</head>  
<body>  
    <div data-role="header" data-position="fixed" data-theme="a">
```

```

        <a href="#" data-rel="back" class="ui-btn ui-btn-left ui-alt-icon ui-nodisc-icon ui-corner-all ui-btn-icon-notext ui-icon-carat-1">Back</a>

        <h1>Fixed external header</h1>
    </div>
<?php } ?>

<div data-role="page" data-title="Details" class="jqm-demos">
    <div role="main" class="ui-content jqm-content jqm-fullwidth">
        <h2>jQuery Mobile</h2>
        <p>The jQuery Mobile is a user interface framework which is built on jQuery core and used for developing responsive websites or applications that are accessible on mobile, tablet and desktop devices. It uses features of both jQuery and jQueryUI to provide API features for mobile web applications.</p>
        <ul class="list">
            <li><p>The jQuery Mobile creates web applications in such a way that it will work the same way on the mobile, tablet and desktop devices.</p></li>
            <li><p>The jQuery Mobile is compatible with other frameworks such as <i>PhoneGap</i>, <i>Whitelight</i> etc.</p></li>
            <li><p>The jQuery Mobile provides set of touch friendly form inputs and UI widgets.</p></li>
            <li><p>The progressive enhancement brings you functionality to all mobile, tablet and desktop platforms and adds efficient page loads and wider device support.</p></li>
        </ul>
    </div>
</div>

<?php if (!isset($_SERVER['HTTP_X_REQUESTED_WITH']) || strtolower($_SERVER['HTTP_X_REQUESTED_WITH']) != 'xmlhttprequest') { ?>
    <div data-role="footer" data-position="fixed" data-theme="a">
        <div data-role="navbar">
            <ul>
                <li><a href="/jquery_mobile/src/toolbar_ajax.html" data-prefetch="true" data-transition="flip">Details</a></li>
                <li><a href="/jquery_mobile/src/toolbar_friends.html" data-prefetch="true" data-transition="fade">Friends</a></li>
                <li><a href="/jquery_mobile/src/toolbar_messages.html" data-prefetch="true" data-transition="slide">Messages</a></li>
            </ul>
        </div>
    </div>
</body>

```

```
</html>
<?php } ?>
```

## Output

Let's carry out the following steps to see how the above code works:

- Save the above html code as **toolbar\_ajax.html** file in your server root folder.
- Open this HTML file as [http://localhost/toolbar\\_ajax.html](http://localhost/toolbar_ajax.html) and the following output will be displayed.

[Back](#)

## Fixed external header

### jQuery Mobile

The jQuery Mobile is a user interface framework which is built on jQuery core and used for developing responsive websites or applications that are accessible on mobile, tablet and desktop devices. It uses features of both jQuery and jQueryUI to provide API features for mobile web applications.

- The jQuery Mobile creates web applications in such a way that it will work the same way on the mobile, tablet and desktop devices.
- The jQuery Mobile is compatible with other frameworks such as *PhoneGap*, *Whitelight* etc.
- The jQuery Mobile provides set of touch friendly form inputs and UI widgets.
- The progressive enhancement brings you functionality to all mobile, tablet and desktop platforms and adds efficient page loads and wider device support.
- [Details](#)
- [Friends](#)
- [Messages](#)

## Form in Toolbar

---

### Description

The form can be used in toolbars. It can be placed in headers and footers using your normal **<form>**.

## Example

Following example demonstrates the use of form in toolbar in jQuery Mobile.

```
<!DOCTYPE html>
<html>
<head>
    <meta name="viewport" content="width=device-width, initial-scale=1">
    <link rel="stylesheet"
    href="https://code.jquery.com/mobile/1.4.5/jquery.mobile-1.4.5.min.css">
    <script src="https://code.jquery.com/jquery-1.11.3.min.js"></script>
    <script src="https://code.jquery.com/mobile/1.4.5/jquery.mobile-
1.4.5.min.js"></script>
</head>
<body>
    <div data-role="page" class="jqm-demos">
        <div data-role="header" data-position="fixed">
            <a href="#" data-rel="back" class="ui-btn ui-btn-left ui-alt-icon ui-
nodisc-icon ui-corner-all ui-btn-icon-notext ui-icon-carat-1">Back</a>
            <h2>Header</h2>
            <form method="post" action="#">
                <fieldset data-role="controlgroup" data-type="horizontal">
                    <legend>Choose your colors:</legend>
                    <label for="red">Red</label>
                    <input type="checkbox" name="favcolor" id="red" value="red">
                    <label for="green">Green</label>
                    <input type="checkbox" name="favcolor" id="green" value="green">
                    <label for="blue">Blue</label>
                    <input type="checkbox" name="favcolor" id="blue" value="blue">
                </fieldset>
            </form>
        </div>
        <div class="ui-content" role="main">
            <ul data-role="listview" data-split-icon="gear" data-theme="a" data-
dividertheme="e" data-filter="true" data-filter-theme="a" data-filter-
placeholder="Search Your Item... ">
                <li><a href="#">
                    <h2>Tshirts</h2></a>
                    <a href="#purchase" data-rel="popup" data-position-to="window"
data-transition="pop">Purchase Item</a>
                </li>
            </ul>
        </div>
    </div>

```

```
</li>
<li><a href="#">
    <h2>Formal Shirts</h2></a>
    <a href="#purchase" data-rel="popup" data-position-to="window"
data-transition="pop">Purchase Item</a>
</li>
<li><a href="#">
    <h2>Trousers</h2></a>
    <a href="#purchase" data-rel="popup" data-position-to="window"
data-transition="pop">Purchase Item</a>
</li>
<li><a href="#">
    <h2>Suits & Blazers</h2></a>
    <a href="#purchase" data-rel="popup" data-position-to="window"
data-transition="pop">Purchase Item</a>
</li>
<li><a href="#">
    <h2>Sneakers</h2></a>
    <a href="#purchase" data-rel="popup" data-position-to="window"
data-transition="pop">Purchase Item</a>
</li>
<li><a href="#">
    <h2>Casual Shoes</h2></a>
    <a href="#purchase" data-rel="popup" data-position-to="window"
data-transition="pop">Purchase Item</a>
</li>
<li><a href="#">
    <h2>Backpacks</h2></a>
    <a href="#purchase" data-rel="popup" data-position-to="window"
data-transition="pop">Purchase Item</a>
</li>
<li><a href="#">
    <h2>Perfumes</h2></a>
    <a href="#purchase" data-rel="popup" data-position-to="window"
data-transition="pop">Purchase Item</a>
</li>
<li><a href="#">
    <h2>Jeans</h2></a>
```

```

        <a href="#purchase" data-rel="popup" data-position-to="window"
data-transition="pop">Purchase Item</a>
    </li>
</ul>

        <div data-role="popup" id="purchase" data-overlay-theme="b" class="ui-
content" style="max-width:340px; padding-bottom:2em;">
            <h2>Purchase Item?</h2>
            <p>In order to purchase this item click Buy or Cancel.</p>
            <a href="#" class="ui-btn ui-btn-b ui-btn-inline ui-mini ui-corner-
all ui-shadow ui-btn-icon-left ui-icon-check" data-rel="back">Buy: $25</a>
            <a href="#" class="ui-btn ui-btn-inline ui-mini ui-corner-all ui-
shadow" data-rel="back">Cancel</a>
        </div>
    </div>
<div data-role="footer" data-theme="a" data-position="fixed">
    <h2>Footer</h2>
    <div class="ui-bar ui-bar-a">
        <form action="#" method="get">
            <input type="text" name="name" id="name" data-mini="true"
placeholder="Name" />
            <input type="email" name="email" id="email" placeholder="Your
email" data-mini="true">
            <textarea name="addinfo" id="info" data-mini="true"
placeholder="Comments"></textarea>
            <button type="submit" class="ui-btn ui-btn-a ui-corner-all ui-
mini">Submit</button>
        </form>
    </div>
</div>
</body>
</html>

```

## Output

Let's carry out the following steps to see how the above code works:

- Save the above html code as **toolbar\_form.html** file in your server root folder.
- Open this HTML file as [http://localhost/toolbar\\_form.html](http://localhost/toolbar_form.html) and the following output will be displayed.

[Back](#)

## Header

Choose your colors:

Red  Green  Blue 

- [Tshirts](#)

[Purchase Item](#)

- [Formal Shirts](#)

[Purchase Item](#)

- [Trousers](#)

[Purchase Item](#)

- [Suits & Blazers](#)

[Purchase Item](#)

- [Sneakers](#)

[Purchase Item](#)

- [Casual Shoes](#)

[Purchase Item](#)

- [Backpacks](#)

[Purchase Item](#)

- [Perfumes](#)

[Purchase Item](#)

- [Jeans](#)

[Purchase Item](#)

## Purchase Item?

In order to purchase this item click Buy or Cancel.

[Buy: \\$25](#) [Cancel](#)

## Footer



## 8. jQuery Mobile – Events

jQuery Mobile allows to create dynamic web pages. By using events, you can set up event-driven process on the elements, which are triggered by the user's interaction such as mouse click, mouse hover on an element, key press on the keyboard, etc.

Following table lists some of the events for the mobile devices, which are supported by jQuery Mobile.

| Sr. No. | Events & Description   |
|---------|--|
| 1       | <b><u>iQuery Mobile Events</u></b><br>It responds to user interaction when the user clicks on a certain page or hovers the mouse over an element, etc. |
| 2       | <b><u>iQuery Touch Events</u></b><br>It provides touch events when the user touches the screen.  |
| 3       | <b><u>iQuery Scroll Events</u></b><br>It fires the scroll events when the user scrolls up and down.  |
| 4       | <b><u>iQuery Orientation Event</u></b><br>It triggers the orientation event when the user rotates the device vertically or horizontally.               |
| 5       | <b><u>iQuery Page Events</u></b><br>It provides the page events when the user hides, creates, loads, or unloads the pages.                             |

### Mobile Events

---

#### Description

An event will respond to user interaction when the user clicks on a certain page or hovers the mouse over an element, etc.

#### Example

Following example describes the use of mobile events in the jQuery Mobile Framework.

```
<!DOCTYPE html>
<head>
<title>Mobile Events</title>
<meta name="viewport" content="width=device-width, initial-scale=1">
```

```
<link rel="stylesheet"
href="https://code.jquery.com/mobile/1.4.5/jquery.mobile-1.4.5.min.css">
<script src="https://code.jquery.com/jquery-1.11.3.min.js"></script>
<script src="https://code.jquery.com/mobile/1.4.5/jquery.mobile-
1.4.5.min.js"></script>
<script>
$(document).on("pagecreate","#page1",function(){
//The on() method attaches the event handlers
//The id #page1 refers to id of the page to specify event
    $("p").on("click",function(){
        $(this).hide();
    });
});
</script>
</head>
<body>
    <div data-role="page" id="page1">
        <div data-role="header">
            <h2>Header</h2>
        </div>

        <div data-role="main" class="ui-content">
            <p>This line will disappear, if you click on it.</p>
        </div>

        <div data-role="footer">
            <h2>Footer</h2>
        </div>
    </div>
</body>
</html>
```

## Output

Let's carry out the following steps to see how the above code works:

- Save the above html code as **jqm\_mobile\_events.html** file in your server root folder.
- Open this HTML file as [http://localhost/jqm\\_mobile\\_events.html](http://localhost/jqm_mobile_events.html) and the following output will be displayed.

## Header

This line will disappear, if you click on it.

## Footer

## Touch Events

---

### Description

It provides touch events when the user touches the screen.

Following table lists some of the touch events supported by jQuery Mobile.

| Sr. No. | Touch Events & Description   |
|---------|--|
| 1       | <b><u>Tap Event</u></b><br>Fires when the user taps on an element.   |
| 2       | <b><u>Taphold Event</u></b><br>Fires when the user taps on an element and holds for a couple of seconds.           |
| 3       | <b><u>Swipe Event</u></b><br>Fires when the user horizontally drags more than 30px over an element.                |
| 4       | <b><u>Swipeleft Event</u></b><br>Fires when the user drags more than 30px over an element in the left direction.   |
| 5       | <b><u>Swiperight Event</u></b><br>Fires when the user drags more than 30px over an element in the right direction. |

## Touch Tap Event

---

### Description

Fires when the user taps on an element using ***id*** of the page to specify an event and ***on()*** method attaches the event handlers.

### Example

Following example describes the use of tap event in jQuery Mobile Framework.

```
<!DOCTYPE html>
<head>
<title>Tap Event</title>
<meta name="viewport" content="width=device-width, initial-scale=1">
<link rel="stylesheet"
href="https://code.jquery.com/mobile/1.4.5/jquery.mobile-1.4.5.min.css">
<script src="https://code.jquery.com/jquery-1.11.3.min.js"></script>
<script src="https://code.jquery.com/mobile/1.4.5/jquery.mobile-
1.4.5.min.js"></script>
<script>
$(document).on("pagecreate","#page1",function(){
    $("p").on("tap",function(){
        $(this).hide();
    });
});
</script>
</head>
<body>
<div data-role="page" id="page1">
    <div data-role="header">
        <h2>Header</h2>
    </div>

    <div data-role="main" class="ui-content">
        <p>This line will disappear, if you tap on it.</p>
    </div>

    <div data-role="footer">
        <h2>Footer</h2>
    </div>
</div>
```

```

    </div>
</div>
</body>
</html>
```

## Output

Let's carry out the following steps to see how the above code works:

- Save the above html code as **jqm\_touch\_tap\_event.html** file in your server root folder.
- Open this HTML file as [http://localhost/jqm\\_touch\\_tap\\_event.html](http://localhost/jqm_touch_tap_event.html) and the following output will be displayed.

## Header

This line will disappear, if you tap on it.

## Footer

## Touch Taphold Event

---

### Description

Fires when the user taps on an element and holds for a couple of seconds using ***id*** of the page to specify an event and ***on()*** method attaches the event handlers.

### Example

Following example describes the use of taphold event in jQuery Mobile Framework.

```

<!DOCTYPE html>
<head>
<title>Taphold Event</title>
<meta name="viewport" content="width=device-width, initial-scale=1">
<link rel="stylesheet"
href="https://code.jquery.com/mobile/1.4.5/jquery.mobile-1.4.5.min.css">
<script src="https://code.jquery.com/jquery-1.11.3.min.js"></script>
<script src="https://code.jquery.com/mobile/1.4.5/jquery.mobile-
1.4.5.min.js"></script>
```

```

<script>
$(document).on("pagecreate","#page1",function(){
    $("p").on("tap",function(){
        $(this).hide();
    });
});
</script>
</head>
<body>
<div data-role="page" id="page1">
    <div data-role="header">
        <h2>Header</h2>
    </div>

    <div data-role="main" class="ui-content">
        <p>This line will disappear, if you tap on it.</p>
    </div>

    <div data-role="footer">
        <h2>Footer</h2>
    </div>
</div>
</body>
</html>

```

## Output

Let's carry out the following steps to see how the above code works:

- Save the above html code as **jqm\_touch\_taphold\_event.html** file in your server root folder.
- Open this HTML file as [http://localhost/jqm\\_touch\\_taphold\\_event.html](http://localhost/jqm_touch_taphold_event.html) and the following output will be displayed.

## Header

This line will disappear, if you tap and hold on it.

## Footer

## Touch Swipe Event

---

### Description

Fires when the user horizontally drags more than 30px over an element using ***id*** of the page to specify an event and ***on()*** method attaches the event handlers.

### Example

Following example describes the use of swipe event in jQuery Mobile Framework.

```
<!DOCTYPE html>
<head>
<title>Swipe Event</title>
<meta name="viewport" content="width=device-width, initial-scale=1">
<link rel="stylesheet"
href="https://code.jquery.com/mobile/1.4.5/jquery.mobile-1.4.5.min.css">
<script src="https://code.jquery.com/jquery-1.11.3.min.js"></script>
<script src="https://code.jquery.com/mobile/1.4.5/jquery.mobile-
1.4.5.min.js"></script>
<script>
$(document).on("pagecreate","#page1",function(){
    $("p").on("swipe",function(){
        $("span").text("swipe event occurred!!!!");
    });
});
</script>
</head>
<body>
<div data-role="page" id="page1">
    <div data-role="header">
```

```

<h2>Header</h2>
</div>

<div data-role="main" class="ui-content">
    <p>It will display the text when you swipe here.</p>
    <span style="color:orange"></span>
</div>

<div data-role="footer">
    <h2>Footer</h2>
</div>
</div>
</body>
</html>

```

## Output

Let's carry out the following steps to see how the above code works:

- Save the above html code as **jqm\_touch\_swipe\_event.html** file in your server root folder.
- Open this HTML file as [http://localhost/jqm\\_touch\\_swipe\\_event.html](http://localhost/jqm_touch_swipe_event.html) and the following output will be displayed.

## Header

It will display the text when you swipe here.

## Footer

## Touch Swipeleft Event

---

### Description

Fires when the user drag more than 30px over an element in the left direction using ***id*** of the page to specify an event and ***on()*** method attaches the event handlers.

## Example

Following example describes the use of swipeleft event in jQuery Mobile Framework.

```
<!DOCTYPE html>
<head>
<title>Swipeleft Event</title>
<meta name="viewport" content="width=device-width, initial-scale=1">
<link rel="stylesheet"
href="https://code.jquery.com/mobile/1.4.5/jquery.mobile-1.4.5.min.css">
<script src="https://code.jquery.com/jquery-1.11.3.min.js"></script>
<script src="https://code.jquery.com/mobile/1.4.5/jquery.mobile-
1.4.5.min.js"></script>
<script>
$(document).on("pagecreate","#page1",function(){
    $("p").on("swipeleft",function(){
        $("span").text("swipe event occurred!!!");
    });
});
</script>
</head>
<body>
<div data-role="page" id="page1">
    <div data-role="header">
        <h2>Header</h2>
    </div>

    <div data-role="main" class="ui-content">
        <p>It will display the text when you swipe left side.</p>
        <span style="color:orange"></span>
    </div>

    <div data-role="footer">
        <h2>Footer</h2>
    </div>
</div>
</body>
</html>
```

## Output

Let's carry out the following steps to see how the above code works:

- Save the above html code as **jqm\_touch\_swipeleft\_event.html** file in your server root folder.
- Open this HTML file as [http://localhost/jqm\\_touch\\_swipeleft\\_event.html](http://localhost/jqm_touch_swipeleft_event.html) and the following output will be displayed.

### Header

It will display the text when you swipe left side.

### Footer

## Touch Swiperight Event

### Description

Fires when the user drags more than 30px over an element in the right direction using ***id*** of the page to specify an event and ***on()*** method attaches the event handlers.

### Example

Following example describes the use of swiperight event in jQuery Mobile Framework.

```
<!DOCTYPE html>
<head>
<title>Swiperight Event</title>
<meta name="viewport" content="width=device-width, initial-scale=1">
<link rel="stylesheet" href="https://code.jquery.com/mobile/1.4.5/jquery.mobile-1.4.5.min.css">
<script src="https://code.jquery.com/jquery-1.11.3.min.js"></script>
<script src="https://code.jquery.com/mobile/1.4.5/jquery.mobile-1.4.5.min.js"></script>
<script>
$(document).on("pagecreate","#page1",function(){
    $("p").on("swiperight",function(){
        $("span").text("swipe event occurred!!!");
    });
});
```

```

});  

</script>  

</head>  

<body>  

<div data-role="page" id="page1">  

  <div data-role="header">  

    <h2>Header</h2>  

  </div>  

  <div data-role="main" class="ui-content">  

    <p>It will display the text when you swipe right side.</p>  

    <span style="color:orange"></span>  

  </div>  

  <div data-role="footer">  

    <h2>Footer</h2>  

  </div>  

</div>  

</body>  

</html>

```

## Output

Let's carry out the following steps to see how the above code works:

- Save the above html code as **jqm\_touch\_swiperight\_event.html** file in your server root folder.
- Open this HTML file as [http://localhost/jqm\\_touch\\_swiperight\\_event.html](http://localhost/jqm_touch_swiperight_event.html) and the following output will be displayed.

## Header

It will display the text when you swipe right side.

## Footer

## Scroll Events

---

### Description

It fires the scroll events when the user scrolls up and down.

Following table lists two types of scroll events, which are supported by jQuery Mobile.

| Sr. No. | Scroll Events & Description   |
|---------|---|
| 1       | <b><u>Scrollstart Event</u></b><br>Fires when the user starts scrolling the page. |
| 2       | <b><u>Scrollstop Event</u></b><br>Fires when the user stops scrolling the page.   |

## Scrollstart Event

---

### Description

Fires when the user starts scrolling the page using ***id*** of the page to specify an event and ***on()*** method attaches the event handlers.

### Example

Following example describes the use of scroll start event in jQuery Mobile Framework.

```
<!DOCTYPE html>
<head>
<title>Scrollstart Event</title>
<meta name="viewport" content="width=device-width, initial-scale=1">
<link rel="stylesheet"
href="https://code.jquery.com/mobile/1.4.5/jquery.mobile-1.4.5.min.css">
<script src="https://code.jquery.com/jquery-1.11.3.min.js"></script>
<script src="https://code.jquery.com/mobile/1.4.5/jquery.mobile-
1.4.5.min.js"></script>
<script>
$(document).on("pagecreate","#page1",function(){
  $(document).on("scrollstart",function(){
    alert("scrolling has been started!!!");
  });
});
</script>
</head>
```

```
<body>
<div data-role="page" id="page1">
    <div data-role="header">
        <h2>Header</h2>
    </div>

    <div data-role="main" class="ui-content">
        <p>Lorem Ipsum is simply dummy text of the printing and typesetting industry. Lorem Ipsum has been the industry's standard dummy text ever since the 1500s, when an unknown printer took a galley of type and scrambled it to make a type specimen book. It has survived not only five centuries, but also the leap into electronic typesetting, remaining essentially unchanged. It was popularised in the 1960s with the release of Letraset sheets containing Lorem Ipsum passages, and more recently with desktop publishing software like Aldus PageMaker including versions of Lorem Ipsum. Lorem Ipsum is simply dummy text of the printing and typesetting industry. Lorem Ipsum has been the industry's standard dummy text ever since the 1500s, when an unknown printer took a galley of type and scrambled it to make a type specimen book. It has survived not only five centuries, but also the leap into electronic typesetting, remaining essentially unchanged. It was popularised in the 1960s with the release of Letraset sheets containing Lorem Ipsum passages, and more recently with desktop publishing software like Aldus PageMaker including versions of Lorem Ipsum. Lorem Ipsum is simply dummy text of the printing and typesetting industry. Lorem Ipsum has been the industry's standard dummy text ever since the 1500s, when an unknown printer took a galley of type and scrambled it to make a type specimen book. It has survived not only five centuries, but also the leap into electronic typesetting, remaining essentially unchanged. It was popularised in the 1960s with the release of Letraset sheets containing Lorem Ipsum passages, and more recently with desktop publishing software like Aldus PageMaker including versions of Lorem Ipsum. Lorem Ipsum is simply dummy text of the printing and typesetting industry. Lorem Ipsum has been the industry's standard dummy text ever since the 1500s, when an unknown printer took a galley of type and scrambled it to make a type specimen book. It has survived not only five centuries, but also the leap into electronic typesetting, remaining essentially unchanged. It was popularised in the 1960s with the release of Letraset sheets containing Lorem Ipsum passages, and more recently with desktop publishing software like Aldus PageMaker including versions of Lorem Ipsum. Lorem Ipsum is simply dummy text of the printing and typesetting industry. Lorem Ipsum has been the industry's standard dummy text ever since the 1500s, when an unknown printer took a galley of type and scrambled it to make a type specimen book. It has survived not only five centuries, but also the leap into electronic typesetting, remaining essentially unchanged. It was popularised in the 1960s with the release of Letraset sheets containing Lorem Ipsum passages, and more recently with desktop publishing software like Aldus PageMaker including versions of Lorem Ipsum. </p>
    </div>
</div>
```

```

<div data-role="footer">
    <h2>Footer</h2>
</div>
</div>
</body>
</html>

```

## Output

Let's carry out the following steps to see how the above code works:

- Save the above html code as **jqm\_scroll\_scrollstart\_event.html** file in your server root folder.
- Open this HTML file as [http://localhost/jqm\\_scroll\\_scrollstart\\_event.html](http://localhost/jqm_scroll_scrollstart_event.html) and the following output will be displayed.

## Header

Lorem Ipsum is simply dummy text of the printing and typesetting industry. Lorem Ipsum has been the industry's standard dummy text ever since the 1500s, when an unknown printer took a galley of type and scrambled it to make a type specimen book. It has survived not only five centuries, but also the leap into electronic typesetting, remaining essentially unchanged. It was popularised in the 1960s with the release of Letraset sheets containing Lorem Ipsum passages, and more recently with desktop publishing software like Aldus PageMaker including versions of Lorem Ipsum. Lorem Ipsum is simply dummy text of the printing and typesetting industry. Lorem Ipsum has been the industry's standard dummy text ever since the 1500s, when an unknown printer took a galley of type and scrambled it to make a type specimen book. It has survived not only five centuries, but also the leap into electronic typesetting, remaining essentially unchanged. It was popularised in the 1960s with the release of Letraset sheets containing Lorem Ipsum passages, and more recently with desktop publishing software like Aldus PageMaker including versions of Lorem Ipsum. Lorem Ipsum is simply dummy text of the printing and typesetting industry. Lorem Ipsum has been the industry's standard dummy text ever since the 1500s, when an unknown printer took a galley of type and scrambled it to make a type specimen book. It has survived not only five centuries, but also the leap into electronic typesetting, remaining essentially unchanged. It was popularised in the 1960s with the release of Letraset sheets containing Lorem Ipsum passages, and more recently with desktop publishing software like Aldus PageMaker including versions of Lorem Ipsum. Lorem Ipsum is simply dummy text of the printing and typesetting industry. Lorem Ipsum has been the industry's standard dummy text ever since the 1500s, when an unknown printer took a galley of type and scrambled it to make a type specimen book. It has survived not only five centuries, but also the leap into electronic typesetting, remaining essentially unchanged. It was popularised in

the 1960s with the release of Letraset sheets containing Lorem Ipsum passages, and more recently with desktop publishing software like Aldus PageMaker including versions of Lorem Ipsum. Lorem Ipsum is simply dummy text of the printing and typesetting industry. Lorem Ipsum has been the industry's standard dummy text ever since the 1500s, when an unknown printer took a galley of type and scrambled it to make a type specimen book. It has survived not only five centuries, but also the leap into electronic typesetting, remaining essentially unchanged. It was popularised in the 1960s with the release of Letraset sheets containing Lorem Ipsum passages, and more recently with desktop publishing software like Aldus PageMaker including versions of Lorem Ipsum. Lorem Ipsum is simply dummy text of the printing and typesetting industry. Lorem Ipsum has been the industry's standard dummy text ever since the 1500s, when an unknown printer took a galley of type and scrambled it to make a type specimen book. It has survived not only five centuries, but also the leap into electronic typesetting, remaining essentially unchanged. It was popularised in the 1960s with the release of Letraset sheets containing Lorem Ipsum passages, and more recently with desktop publishing software like Aldus PageMaker including versions of Lorem Ipsum.

## Footer

## Scrollstop Event

### Description

Fires when the user stops scrolling the page using ***id*** of the page to specify an event and ***on()*** method attaches the event handlers.

### Example

Following example describes the use of scroll stop event in jQuery Mobile Framework.

```
<!DOCTYPE html>
<head>
<title>Tap Event</title>
<meta name="viewport" content="width=device-width, initial-scale=1">
<link rel="stylesheet"
href="https://code.jquery.com/mobile/1.4.5/jquery.mobile-1.4.5.min.css">
<script src="https://code.jquery.com/jquery-1.11.3.min.js"></script>
<script src="https://code.jquery.com/mobile/1.4.5/jquery.mobile-
1.4.5.min.js"></script>
<script>
$(document).on("pagecreate","#page1",function(){
$(document).on("scrollstop",function(){
alert("scrolling has been stopped!!!");
});
});
```



```

with desktop publishing software like Aldus PageMaker including versions of Lorem
Ipsum.</p>
</div>

<div data-role="footer">
    <h2>Footer</h2>
</div>
</div>
</body>
</html>

```

## Output

Let's carry out the following steps to see how the above code works:

- Save the above html code as **jqm\_scroll\_scrollstop\_event.html** file in your server root folder.
- Open this HTML file as [http://localhost/jqm\\_scroll\\_scrollstop\\_event.html](http://localhost/jqm_scroll_scrollstop_event.html) and the following output will be displayed.

## Header

Lorem Ipsum is simply dummy text of the printing and typesetting industry. Lorem Ipsum has been the industry's standard dummy text ever since the 1500s, when an unknown printer took a galley of type and scrambled it to make a type specimen book. It has survived not only five centuries, but also the leap into electronic typesetting, remaining essentially unchanged. It was popularised in the 1960s with the release of Letraset sheets containing Lorem Ipsum passages, and more recently with desktop publishing software like Aldus PageMaker including versions of Lorem Ipsum. Lorem Ipsum is simply dummy text of the printing and typesetting industry. Lorem Ipsum has been the industry's standard dummy text ever since the 1500s, when an unknown printer took a galley of type and scrambled it to make a type specimen book. It has survived not only five centuries, but also the leap into electronic typesetting, remaining essentially unchanged. It was popularised in the 1960s with the release of Letraset sheets containing Lorem Ipsum passages, and more recently with desktop publishing software like Aldus PageMaker including versions of Lorem Ipsum. Lorem Ipsum is simply dummy text of the printing and typesetting industry. Lorem Ipsum has been the industry's standard dummy text ever since the 1500s, when an unknown printer took a galley of type and scrambled it to make a type specimen book. It has survived not only five centuries, but also the leap into electronic typesetting, remaining essentially unchanged. It was popularised in the 1960s with the release of Letraset sheets containing Lorem Ipsum passages, and more recently with desktop publishing software like Aldus PageMaker including versions of Lorem Ipsum. Lorem Ipsum is simply dummy text of the printing and typesetting industry. Lorem Ipsum has been the industry's standard

dummy text ever since the 1500s, when an unknown printer took a galley of type and scrambled it to make a type specimen book. It has survived not only five centuries, but also the leap into electronic typesetting, remaining essentially unchanged. It was popularised in the 1960s with the release of Letraset sheets containing Lorem Ipsum passages, and more recently with desktop publishing software like Aldus PageMaker including versions of Lorem Ipsum. Lorem Ipsum is simply dummy text of the printing and typesetting industry. Lorem Ipsum has been the industry's standard dummy text ever since the 1500s, when an unknown printer took a galley of type and scrambled it to make a type specimen book. It has survived not only five centuries, but also the leap into electronic typesetting, remaining essentially unchanged. It was popularised in the 1960s with the release of Letraset sheets containing Lorem Ipsum passages, and more recently with desktop publishing software like Aldus PageMaker including versions of Lorem Ipsum. Lorem Ipsum is simply dummy text of the printing and typesetting industry. Lorem Ipsum has been the industry's standard dummy text ever since the 1500s, when an unknown printer took a galley of type and scrambled it to make a type specimen book. It has survived not only five centuries, but also the leap into electronic typesetting, remaining essentially unchanged. It was popularised in the 1960s with the release of Letraset sheets containing Lorem Ipsum passages, and more recently with desktop publishing software like Aldus PageMaker including versions of Lorem Ipsum.

## Footer

## Orientation Event

### Description

You can trigger the orientation event when the user rotates the device vertically or horizontally using ***orientationchange*** event. It uses the ***window.orientation*** property to specify whether the window or the device is set to vertical or horizontal orientation.

### Example

Following example describes the use of orientation event in jQuery Mobile Framework.

```
<!DOCTYPE html>
<head>
<title>Orientation Event</title>
<meta name="viewport" content="width=device-width, initial-scale=1">
<link rel="stylesheet"
href="https://code.jquery.com/mobile/1.4.5/jquery.mobile-1.4.5.min.css">
<script src="https://code.jquery.com/jquery-1.11.3.min.js"></script>
<script src="https://code.jquery.com/mobile/1.4.5/jquery.mobile-
1.4.5.min.js"></script>
<script>
```

```

$(document).on("pagecreate","#page1",function(event){
    $(window).on("orientationchange",function(){
        if(window.orientation == 0)
        {
            $("p").text("portrait orientation!!!");
        }
        else
        {
            $("p").text("landscape orientation!!!");
        }
    });
});
</script>
</head>
<body>
    <div data-role="page" id="page1">
        <div data-role="header">
            <h2>Header</h2>
        </div>

        <div data-role="main" class="ui-content">
            <p>To see the orientation effect, you must rotate your window...</p>
        </div>

        <div data-role="footer">
            <h2>Footer</h2>
        </div>
    </div>
</body>
</html>

```

## Output

Let's carry out the following steps to see how the above code works:

- Save the above html code as **jqm\_orientation\_event.html** file in your server root folder.
- Open this HTML file as [http://localhost/jqm\\_orientation\\_event.html](http://localhost/jqm_orientation_event.html) and the following output will be displayed.

## Header

To see the orientation effect, you must rotate your window...

## Footer

## Page Events

---

### Description

It provides the page events when the user hides, creates, loads, or unloads the pages.

Following table lists two types of page events, which are supported by jQuery Mobile.

| Sr. No. | Page Events & Description  |
|---------|--|
| 1       | <b><u>Initialization Events</u></b><br>Fires when the page has been initialized.         |
| 2       | <b><u>Load Events</u></b><br>Fires when the external pages are loaded into the DOM.      |
| 3       | <b><u>Transition Events</u></b><br>Fires when we transition from one page to other page. |

## Page Initialization Events

---

### Description

Fires when the page has been initialized using two types of events:

- The **pagebeforecreate** event fires when the page is about to be initialized.
- The **pagecreate** event fires when the page has been created.

## Example

Following example describes the use of page initialization events in jQuery Mobile Framework.

```
<!DOCTYPE html>
<head>
<title>Initialization Event</title>
<meta name="viewport" content="width=device-width, initial-scale=1">
<link rel="stylesheet"
href="https://code.jquery.com/mobile/1.4.5/jquery.mobile-1.4.5.min.css">
<script src="https://code.jquery.com/jquery-1.11.3.min.js"></script>
<script src="https://code.jquery.com/mobile/1.4.5/jquery.mobile-
1.4.5.min.js"></script>
<script>
$(document).on("pagebeforecreate",function(){
    alert("The pagebeforecreate event has been fired and the page is not yet
initialized.");
});
$(document).on("pagecreate",function(){
    alert("The pagecreate event has been fired and the page has been
created.");
});
</script>
</head>
<body>
<div data-role="page" id="page1">
    <div data-role="header">
        <h2>Header</h2>
    </div>

    <div data-role="main" class="ui-content">
        <p>The page has been initialized.</p>
    </div>

    <div data-role="footer">
        <h2>Footer</h2>
    </div>
</div>
</body>
```

```
</html>
```

## Output

Let's carry out the following steps to see how the above code works:

- Save the above html code as **jqm\_initialization\_event.html** file in your server root folder.
- Open this HTML file as [http://localhost/jqm\\_page\\_initialization\\_event.html](http://localhost/jqm_page_initialization_event.html) and the following output will be displayed.

## Header

The page has been initialized.

## Footer

## Page Load Events

---

### Description

Fires when the page has been initialized using two types of events:

- The **pagebeforecreate** event fires when the page is about to be initialized.
- The **pagecreate** event fires when the page has been created.

### Description

It fires when the external pages are loaded into the DOM using the following types of events:

- The **pagecontainerbeforeload** event fires before page load.
- The **pagecontainerload** event fires when the page has been loaded and placed in the DOM.
- The **pagecontainerloadfailed** event fires when the page fails to load and displays the "Error Loading Page" message.

### Example

Following example describes the use of page load events in jQuery Mobile Framework.

```
<!DOCTYPE html>
<head>
<title>Page Load Events</title>
<meta name="viewport" content="width=device-width, initial-scale=1">
<link rel="stylesheet"
href="https://code.jquery.com/mobile/1.4.5/jquery.mobile-1.4.5.min.css">
<script src="https://code.jquery.com/jquery-1.11.3.min.js"></script>
<script src="https://code.jquery.com/mobile/1.4.5/jquery.mobile-
1.4.5.min.js"></script>
<script>
$(document).on("pagecontainerload",function(event,data){
    alert("The page has been loaded: " + data.url);
});
$(document).on("pagecontainerloadfailed",function(event,data){
    alert("Sorry, the page has been failed to load!!!!");
});
</script>
</head>
<body>
<div data-role="page" id="page1">
    <div data-role="header">
        <h2>Header</h2>
    </div>

    <div data-role="main" class="ui-content">
        <a href="mypage.html">My Page</a>
    </div>

    <div data-role="footer">
        <h2>Footer</h2>
    </div>
</div>
</body>
</html>
```

## Output

Let's carry out the following steps to see how the above code works:

- Save the above html code as **jqm\_page\_load\_event.html** file in your server root folder.
- Open this HTML file as [http://localhost/jqm\\_page\\_load\\_event.html](http://localhost/jqm_page_load_event.html) and the following output will be displayed.

## Header

[My Page](#)

## Footer

## Page Transition Events

---

### Description

Fires when we transition from one page to other using the following event types:

- The **pagebeforeshow** event fires when triggering on the "toPage" before the transition animation starts.
- The **pageshow** event fires when triggering on the "toPage" after the transition animation occurs.
- The **pagebeforehide** event fires when triggering on the "fromPage" before the transition animation starts.
- The **pagehide** event fires when triggering on the "fromPage" after the transition animation occurs.

### Example

Following example describes the use of page transition events in jQuery Mobile Framework.

```
<!DOCTYPE html>
<head>
<title>Page Transition Events</title>
<meta name="viewport" content="width=device-width, initial-scale=1">
<link rel="stylesheet"
href="https://code.jquery.com/mobile/1.4.5/jquery.mobile-1.4.5.min.css">
<script src="https://code.jquery.com/jquery-1.11.3.min.js"></script>
```

```
<script src="https://code.jquery.com/mobile/1.4.5/jquery.mobile-1.4.5.min.js"></script>

<script>
    $(document).on("pagebeforeshow","#page2",function(){
        alert("The pagebeforeshow event has been fired and second page is yet to be shown.");
    });

    $(document).on("pageshow","#page2",function(){
        alert("The pageshow event has been fired and second page is displaying now.");
    });

    $(document).on("pagebeforehide","#page2",function(){
        alert("The pagebeforehide event has been fired and second page is going to be hide now.");
    });

    $(document).on("pagehide","#page2",function(){
        alert("The pagehide event has been fired and second page is hidden now.");
    });
</script>
</head>
<body>
<div data-role="page" id="page1">
    <div data-role="header">
        <h2>Header</h2>
    </div>

    <div data-role="main" class="ui-content">
        <p>This is first page</p>
        <a href="#page2">Page Two</a>
    </div>

    <div data-role="footer">
        <h2>Footer</h2>
    </div>
</div>

<div data-role="page" id="page2">
    <div data-role="header">
        <h2>Header</h2>
```

```

</div>

<div data-role="main" class="ui-content">
    <p>This is second page</p>
    <a href="#page1">Page One</a>
</div>

<div data-role="footer">
    <h2>Footer</h2>
</div>
</div>
</body>
</html>

```

## Output

Let's carry out the following steps to see how the above code works:

- Save the above html code as **jqm\_page\_transition\_event.html** file in your server root folder.
- Open this HTML file as [http://localhost/jqm\\_page\\_transition\\_events.html](http://localhost/jqm_page_transition_events.html) and the following output will be displayed.

## Header

This is first page

[Page Two](#)

## Footer

## Header

This is second page

[Page One](#)

## Footer

## 9. jQuery Mobile – Forms

Creation of forms is easy and very flexible, which are built with a combination of standardized form elements and buttons.

Following table demonstrates the types of form in detail.

| Sr. No. | Types & Description   |
|---------|---|
| 1       | <b><u>Form Basic</u></b><br>jQuery Mobile provides powerful, easy, and versatile layout system for Forms, which combines form styles, input button, and slider support.                     |
| 2       | <b><u>Form Inputs</u></b><br>The <input> tag is a control that allows the user to input data.   |
| 3       | <b><u>Form Select</u></b><br>In form of option, a dropdown list is provided for select menu.  |
| 4       | <b><u>Form Sliders</u></b><br>Slider allows you to choose a value by sliding the handle of the slider.  |
| 5       | <b><u>Refreshing and auto initialization of form elements</u></b><br>Refresh method is used to update the new state of form control by itself and updates the form control with JavaScript. |

### Form Basic

#### Description

jQuery Mobile provides a powerful, easy, and versatile layout system for Forms, which combines form styles, input button and slider support.

Following table demonstrates the types of basic form in detail.

| Sr. No. | Types & Description   |
|---------|---|
| 1       | <b><u>Form Structure</u></b><br>It is used to create a form layout to collect user input. |
| 2       | <b><u>Form Buttons</u></b><br>The <input> element is used to code button in forms.        |

3

**Form Field Containers**

*ui-field-contain* class is used to wrap the label and input in the form.

## Form Structure

### Description

It is used to create a form layout to collect user input. The **<form>** element should contain action and method attribute that submit via HTTP POST or GET. Following form controls can be used to create the structure.

- Text Input
- Select Input
- Checkbox Input
- Radio Input
- Slider Input
- Search Input

### Example

Following example demonstrates the use of form structure in jQuery Mobile.

```
<!DOCTYPE html>
<html>
<head>
    <meta name="viewport" content="width=device-width, initial-scale=1">
    <link rel="stylesheet"
    href="https://code.jquery.com/mobile/1.4.5/jquery.mobile-1.4.5.min.css">
    <script src="https://code.jquery.com/jquery-1.11.3.min.js"></script>
    <script src="https://code.jquery.com/mobile/1.4.5/jquery.mobile-
1.4.5.min.js"></script>
</head>
<body>
<div data-role="page">
    <div data-role="header">
        <h2>Form Structure</h2>
    </div>
    <div data-role="main" class="ui-content">
        <form method="post" action="demo.php">
            <label for="fname">Name</label>
            <input type="text" name="fname" id="fname" placeholder="Full Name">
    </div>
</div>

```

```
<label for="date">Date</label>
<input type="date" name="date" id="date">

<label for="select">Select City</label>
<select name="select" id="select">
    <option value="1">Belgaum</option>
    <option value="2">Pune</option>
    <option value="3">Chennai</option>
    <option value="4">Bangalore</option>
    <option value="5">Mumbai</option>
</select>

Flipswitch
<input type="checkbox" data-role="flipswitch"><br/>

Gender
<label for="radio1">
    <input type="radio" name="radio-choice-0" id="radio1">Male</input>
</label>
<label for="radio2">
    <input type="radio" name="radio-choice-0" id="radio2">Female</input>
</label>
<label for="radio3">
    <input type="radio" name="radio-choice-0" id="radio3">Other</input>
</label>

Education Qualification
<label for="checkbox1">
    <input type="checkbox" id="checkbox1">BE
</label>
<label for="checkbox2">
    <input type="checkbox" id="checkbox2">BCA
</label>
<label for="checkbox3">
    <input type="checkbox" id="checkbox3">BBA
</label>
```

```

<label for="checkbox4">
    <input type="checkbox" id="checkbox4">MBA
</label>
<label for="checkbox5">
    <input type="checkbox" id="checkbox5">MCA
</label>
</form>
</div>
</div>
</body>
</html>

```

## Output

Let's carry out the following steps to see how the above code works:

- Save the above html code as **form\_structure.html** file in your server root folder.
- Open this HTML file as [http://localhost/form\\_structure.html](http://localhost/form_structure.html) and the following output will be displayed.

## Form Structure

Name  Date  Select City  ▾

Flipswitch

Gender  Male  Female  Other Education Qualification  BE  BCA  BBA  MBA  MCA

## Form Buttons

---

### Description

jQuery Mobile supports button styles that can be customized according to your needs. The **<input>** element is used to code button in forms. Following are the attributes used to modify the style of buttons.

| Sr. No. | Attribute    | Value                        | Description                                   |
|---------|--------------|------------------------------|---|
| 1       | data-corners | true/false                   | Sets the corner for the buttons.              |
| 2       | data-icon    | <b>Different icons</b>       | Sets different icon on the button.            |
| 3       | data-iconpos | left/right/bottom/top/notext | Sets the position for the icon on the button. |
| 4       | data-inline  | true/false                   | Sets the button inline.                       |
| 5       | data-mini    | true/false                   | Sets the button small.                        |
| 6       | data-shadow  | true/false                   | Sets the shadow for the button.               |

## jQuery Mobile—Icon Set

---

### Description

You can set the icon in the button using **ui-icon** class and **ui-btn-icon-pos\_name** class for specifying the position for the icon.

### Example

Following example describes the use of icon setting in jQuery Mobile Framework.

```
<!DOCTYPE html>
<head>
<meta name="viewport" content="width=device-width, initial-scale=1">
<link rel="stylesheet"
href="https://code.jquery.com/mobile/1.4.5/jquery.mobile-1.4.5.min.css">
<script src="https://code.jquery.com/jquery-1.11.3.min.js"></script>
<script src="https://code.jquery.com/mobile/1.4.5/jquery.mobile-
1.4.5.min.js"></script>
</head>
<body>
<div data-role="main" class="ui-content">
<a href="#" class="ui-btn ui-btn-icon-left ui-icon-grid">Grid Button</a>
<button class="ui-btn ui-btn-icon-left ui-icon-arrow-d-r">arrow-d-r</button>

```

```

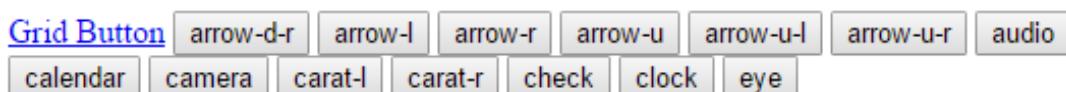
<button class="ui-btn ui-btn-icon-left ui-icon-arrow-l">arrow-l</button>
<button class="ui-btn ui-btn-icon-left ui-icon-arrow-r">arrow-r</button>
<button class="ui-btn ui-btn-icon-left ui-icon-arrow-u">arrow-u</button>
<button class="ui-btn ui-btn-icon-left ui-icon-arrow-u-l">arrow-u-l</button>
<button class="ui-btn ui-btn-icon-left ui-icon-arrow-u-r">arrow-u-r</button>
<button class="ui-btn ui-btn-icon-left ui-icon-audio">audio</button>
<button class="ui-btn ui-btn-icon-left ui-icon-calendar">calendar</button>
<button class="ui-btn ui-btn-icon-left ui-icon-camera">camera</button>
<button class="ui-btn ui-btn-icon-left ui-icon-carat-l">carat-l</button>
<button class="ui-btn ui-btn-icon-left ui-icon-carat-r">carat-r</button>
<button class="ui-btn ui-shadow ui-corner-all ui-btn-icon-left ui-icon-check">check</button>
    <button class="ui-btn ui-shadow ui-corner-all ui-btn-icon-left ui-icon-clock">clock</button>
        <button class="ui-btn ui-shadow ui-corner-all ui-btn-icon-left ui-icon-eye">eye</button>
</div>
</body>
</html>

```

## Output

Let's carry out the following steps to see how the above code works:

- Save the above html code as **icons\_icon\_set.html** file in your server root folder.
- Open this HTML file as [http://localhost/icons\\_icon\\_set.html](http://localhost/icons_icon_set.html) and the following output will be displayed.



## Example

Following example demonstrates the use of form buttons in jQuery Mobile.

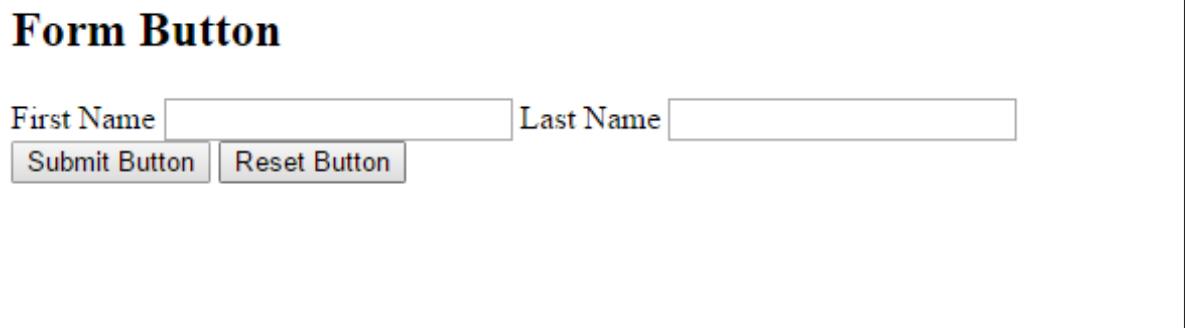
```
<!DOCTYPE html>
<html>
<head>
    <meta name="viewport" content="width=device-width, initial-scale=1">
    <link rel="stylesheet"
    href="https://code.jquery.com/mobile/1.4.5/jquery.mobile-1.4.5.min.css">
    <script src="https://code.jquery.com/jquery-1.11.3.min.js"></script>
    <script src="https://code.jquery.com/mobile/1.4.5/jquery.mobile-
1.4.5.min.js"></script>
</head>
<body>
<div data-role="page">
    <div data-role="header">
        <h2>Form Button</h2>
    </div>
    <div data-role="main" class="ui-content">
        <form method="post" action="jquery_mobile/demo.php">
            <label for="fname">First Name</label>
            <input type="text" name="fname">
            <label for="lname">Last Name</label>
            <input type="text" name="lname">

            <input type="submit" value="Submit Button" data-inline="true" data-
corner="true">
            <input type="reset" value="Reset Button" data-inline="true" data-
corner="true">
        </form>
    </div>
</div>
</body>
</html>
```

## Output

Let's carry out the following steps to see how the above code works:

- Save the above html code as **form\_buttons.html** file in your server root folder.
- Open this HTML file as [http://localhost/form\\_buttons.html](http://localhost/form_buttons.html) and the following output will be displayed.



## Form Button

First Name  Last Name

## Form Field Container

---

### Description

**ui-field-contain** class is used to wrap the label and input in the form.

### Example

Following example demonstrates the use of form field container in jQuery Mobile.

```
<!DOCTYPE html>
<html>
<head>
    <meta name="viewport" content="width=device-width, initial-scale=1">
    <link rel="stylesheet"
    href="https://code.jquery.com/mobile/1.4.5/jquery.mobile-1.4.5.min.css">
    <script src="https://code.jquery.com/jquery-1.11.3.min.js"></script>
    <script src="https://code.jquery.com/mobile/1.4.5/jquery.mobile-
1.4.5.min.js"></script>
</head>
<body>
<div data-role="page">
    <div data-role="header">
        <h2>Form Field contain</h2>
    </div>
    <div data-role="main" class="ui-content">
```

```
<form method="post" action="jquery_mobile/demo.php">
    <div class="ui-field-contain">
        <label for="fname">First Name</label>
        <input type="text" name="fname">
        <label for="lname">Last Name</label>
        <input type="text" name="lname">

        <label for="select">Select City</label>
        <select name="select" id="select">
            <option value="1">Belgaum</option>
            <option value="2">Pune</option>
            <option value="3">Chennai</option>
            <option value="4">Bangalore</option>
            <option value="5">Mumbai</option>
        </select>

        <input type="submit" value="Submit" data-inline="true">
    </div>
</form>
</div>
</body>
</html>
```

## Output

Let's carry out the following steps to see how the above code works:

- Save the above html code as **form\_field\_container.html** file in your server root folder.
- Open this HTML file as [http://localhost/form\\_field\\_container.html](http://localhost/form_field_container.html) and the following output will be displayed.

## Form Field contain

First Name  Last Name  Select City

## Form Inputs

### Description

The **<input>** tag is a control that allows the user to input data. There are different types of input present, i.e. text, search, email, date, checkbox, radio, and so on.

### Example

Following example demonstrates the use of form inputs in jQuery Mobile.

```
<!DOCTYPE html>
<html>
<head>
    <meta name="viewport" content="width=device-width, initial-scale=1">
    <link rel="stylesheet"
    href="https://code.jquery.com/mobile/1.4.5/jquery.mobile-1.4.5.min.css">
    <script src="https://code.jquery.com/jquery-1.11.3.min.js"></script>
    <script src="https://code.jquery.com/mobile/1.4.5/jquery.mobile-
1.4.5.min.js"></script>
</head>
<body>
<div data-role="page">
    <div data-role="header">
        <h2>Form Inputs</h2>
        <input type="search" name="src">
    </div>
    <div data-role="main" class="ui-content">
        <form method="post" action="jquery_mobile/demo1.php">
            <label for="fname">First Name</label>
            <input type="text" name="fname">
    </div>
</div>

```

```
<label for="email">Email Address</label>
<input type="email" name="email">

<label for="date">Birth Date</label>
<input type="date" name="date">

<label for="file">Upload File</label>
<input type="file" name="file">

Select Car
<fieldset data-role="controlgroup" name="checkbox">
    <input type="checkbox" name="checkbox" id="checkbox">
    <label for="checkbox">BMW</label>

    <input type="checkbox" name="checkbox" id="checkbox1">
    <label for="checkbox1">Audi</label>

    <input type="checkbox" name="checkbox" id="checkbox2">
    <label for="checkbox2">Skoda</label>
</fieldset>

Gender
<fieldset data-role="controlgroup">
    <input type="radio" name="radio" id="radio">
    <label for="radio">Male</label>

    <input type="radio" name="radio" id="radio1">
    <label for="radio1">Female</label>
</fieldset>

<label for="slider">Range</label>
<input type="range" name="slider" id="slider" value="20" min="0" max="100" data-highlight="true">

<input type="submit" value="Submit" data-inline="true">
</form>
```

```

</div>
</div>
</body>
</html>

```

## Output

Let's carry out the following steps to see how the above code works:

- Save the above html code as **form\_inputs.html** file in your server root folder.
- Open this HTML file as [http://localhost/form\\_inputs.html](http://localhost/form_inputs.html) and the following output will be displayed.

## Form Inputs

The screenshot shows a mobile-style form titled "Form Inputs". It contains the following elements:

- Text input field for First Name.
- Text input field for Email Address.
- Date input field for Birth Date (dd - yyyy).
- File input field for Upload File, showing "No file chosen".
- Select input field for Select Car, with options BMW, Audi, and Skoda.
- Text input field for Gender, with radio buttons for Male and Female.
- Range input field with a slider.
- Submit button.

## Form Select

### Description

In form, a dropdown list is provided for select menu.

Following table demonstrates the types of basic form in detail.

| Sr. No. | Types & Description  |
|---------|--|
| 1       | <b>Select Menus</b><br>Select menu creates a simple menu with select options to choose from.                           |
| 2       | <b>Custom Select Menus</b><br>The custom select menu is used to display the menu list in the popup format in the form. |

3

**Multiple Selection**

Include `multiple="multiple"` in the `<select>` elements to make multiple selected option in the select menu.

## Form Select Menus

---

### Description

Select menu creates a simple menu with select options to choose from. The `<select></select>` elements are used to define a selection list and the `<option></option>` tags are used to define an item in a selection list.

### Example

Following example demonstrates the use of form select menus in the jQuery Mobile.

```
<!DOCTYPE html>
<html>
<head>
    <meta name="viewport" content="width=device-width, initial-scale=1">
    <link rel="stylesheet"
    href="https://code.jquery.com/mobile/1.4.5/jquery.mobile-1.4.5.min.css">
    <script src="https://code.jquery.com/jquery-1.11.3.min.js"></script>
    <script src="https://code.jquery.com/mobile/1.4.5/jquery.mobile-
1.4.5.min.js"></script>
</head>
<body>
<div data-role="page">
    <div data-role="header">
        <h2>Form Select Menus</h2>
    </div>
    <div data-role="main" class="ui-content">
        <form method="post" action="jquery_mobile/demo.php">
            <fieldset class="ui-field-contain">
                <label for="fname">Form Select Menu</label>
                <select id="select" name="fname">
                    <option value="Pune">Pune</option>
                    <option value="Belgaum">Belgaum</option>
                    <option value="Chennai">Chennai</option>
                    <option value="Bangalore">Bangalore</option>
                </select>
            </fieldset>
        </form>
    </div>
</div>

```

```

</fieldset>
<input type="submit" value="Submit" data-inline="true">
</form>
</div>
</div>
</body>
</html>

```

## Output

Let's carry out the following steps to see how the above code works:

- Save the above html code as **form\_select\_menus.html** file in your server root folder.
- Open this HTML file as [http://localhost/form\\_select\\_menus.html](http://localhost/form_select_menus.html) and the following output will be displayed.



## Form Select Menus

Form Select Menu Pune ▾

Submit

## Form Custom Select Menus

### Description

The custom select menu is used to display the menu list in the popup format in the form.

### Example

Following example demonstrates the use of form custom select menus in jQuery Mobile.

```

<!DOCTYPE html>
<html>
<head>
    <meta name="viewport" content="width=device-width, initial-scale=1">
    <link rel="stylesheet"
    href="https://code.jquery.com/mobile/1.4.5/jquery.mobile-1.4.5.min.css">
    <script src="https://code.jquery.com/jquery-1.11.3.min.js"></script>

```

```

<script src="https://code.jquery.com/mobile/1.4.5/jquery.mobile-
1.4.5.min.js"></script>

</head>

<body>

<div data-role="page">
    <div data-role="header">
        <h2>Form Custom Select Menus</h2>
    </div>
    <div data-role="main" class="ui-content">
        <form method="post" action="jquery_mobile/demo.php">
            <fieldset class="ui-field-contain">
                <label for="select-custom">Custom Select</label>
                <select id="select-custom" data-native-menu="false" name="fname">
                    <option value="Mumbai">Mumbai</option>
                    <option value="Pune">Pune</option>
                    <option value="Belgaum">Belgaum</option>
                    <option value="Chennai">Chennai</option>
                    <option value="Bangalore">Bangalore</option>
                </select>
            </fieldset>
            <input type="submit" value="Submit" data-inline="true">
        </form>
    </div>
</div>
</body>
</html>

```

## Output

Let's carry out the following steps to see how the above code works:

- Save the above html code as **form\_custom\_select\_menus.html** file in your server root folder.
- Open this HTML file as [http://localhost/form\\_custom\\_select\\_menus.html](http://localhost/form_custom_select_menus.html) and the following output will be displayed.

## Form Custom Select Menus

Custom Select Mumbai ▾

Submit

## Form Multiple Selection

### Description

Include ***multiple="multiple"*** in the **<select>** elements to make multiple selected option in the select menu.

### Example

Following example demonstrates the use of multiple selection in form in jQuery Mobile.

```
<!DOCTYPE html>
<html>
<head>
    <meta name="viewport" content="width=device-width, initial-scale=1">
    <link rel="stylesheet"
    href="https://code.jquery.com/mobile/1.4.5/jquery.mobile-1.4.5.min.css">
    <script src="https://code.jquery.com/jquery-1.11.3.min.js"></script>
    <script src="https://code.jquery.com/mobile/1.4.5/jquery.mobile-
1.4.5.min.js"></script>
</head>
<body>
<div data-role="page">
    <div data-role="header">
        <h2>Form Multiple Selection</h2>
    </div>
    <div data-role="main" class="ui-content">
        <form method="get" action="jquery_mobile/demo2.php">
            <fieldset class="ui-field-contain">
                <label for="test">Multiple Select</label>

```

```

<select id="test" multiple="multiple" data-native-menu="false"
name="test[]>
    <option value="Mumbai">Mumbai</option>
    <option value="Pune">Pune</option>
    <option value="Belgaum">Belgaum</option>
    <option value="Chennai">Chennai</option>
    <option value="Bangalore">Bangalore</option>
</select>
</fieldset>
<input type="submit" value="Submit" name="submit" data-inline="true">
</form>
</div>
</div>
</body>
</html>
```

## Output

Let's carry out the following steps to see how the above code works:

- Save the above html code as **form\_multiple\_selection.html** file in your server root folder.
- Open this HTML file as [http://localhost/form\\_multiple\\_selection.html](http://localhost/form_multiple_selection.html) and the following output will be displayed.

## Form Multiple Selection

Mumbai  
Pune  
Belgaum  
Chennai

Multiple Select

Submit

## Form Slider

---

### Description

Slider allows you to choose a value by sliding the handle of the slider.

Following table demonstrates the types of slider in form in detail.

| Sr. No. | Types & Description  |
|---------|--|
| 1       | <b><u>Slider Controls</u></b><br>Slider allows you to choose a numeric value from a range of value.  |
| 2       | <b><u>Flip Toggle Switch</u></b><br>Flip Switch allows you to turn off/on or true/false the switch by clicking it for boolean style input. |

## Form Slider Controls

---

### Description

Slider allows you to choose a numeric value from a range of values. Slider provides you a handle to slide and select a value.

### Example

Following example demonstrates the use of form slider control in jQuery Mobile.

```
<!DOCTYPE html>
<html>
<head>
  <meta name="viewport" content="width=device-width, initial-scale=1">
  <link rel="stylesheet"
  href="https://code.jquery.com/mobile/1.4.5/jquery.mobile-1.4.5.min.css">
  <script src="https://code.jquery.com/jquery-1.11.3.min.js"></script>
  <script src="https://code.jquery.com/mobile/1.4.5/jquery.mobile-
1.4.5.min.js"></script>
</head>
<body>
<div data-role="page">
  <div data-role="header">
    <h2>Form Multiple Selection</h2>
  </div>
```

```

<div data-role="main" class="ui-content">
    <form method="POST" action="sliderdemo.php">
        <label for="slider1">Simple Slider</label>
        <input type="range" name="slider1" id="slider1" min="0" max="100"
value="60">

        <label for="range-1">Show Value:</label>
        <input type="range" name="range-1" id="range-1" min="0" max="100"
value="20" data-show-value="true">

        <label for="range-2">Popup Value:</label>
        <input type="range" name="range-2" id="range-2" min="0" max="100"
value="60" data-popup-enabled="true">

        <label for="range-3">Highlight slider:</label>
        <input type="range" name="range-3" id="range-3" min="0" max="100"
value="30" data-highlight="true">

        <input type="submit" value="Submit" name="submit" data-inline="true">
    </form>
</div>
</div>
</body>
</html>

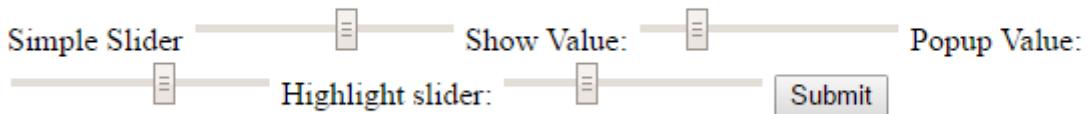
```

## Output

Let's carry out the following steps to see how the above code works:

- Save the above html code as **form\_slider\_controls.html** file in your server root folder.
- Open this HTML file as [http://localhost/form\\_slider\\_controls.html](http://localhost/form_slider_controls.html) and the following output will be displayed.

## Form Multiple Selection



## Form Flip Toggle Switch

### Description

Flip Switch allows you to turn off/on or true/false the switch by clicking it for boolean style input.

### Example

Following example demonstrates the use of multiple selection in form in jQuery Mobile.

```
<!DOCTYPE html>
<html>
<head>
    <meta name="viewport" content="width=device-width, initial-scale=1">
    <link rel="stylesheet"
    href="https://code.jquery.com/mobile/1.4.5/jquery.mobile-1.4.5.min.css">
    <script src="https://code.jquery.com/jquery-1.11.3.min.js"></script>
    <script src="https://code.jquery.com/mobile/1.4.5/jquery.mobile-
1.4.5.min.js"></script>
</head>
<body>
<div data-role="page">
    <div data-role="header">
        <h2>Form Multiple Selection</h2>
    </div>
    <div data-role="main" class="ui-content">
        <form method="post" action="switchdemo.php">
            <h3>Basic Flipswitch</h3>
            <input name="flip" type="checkbox" data-role="flipswitch">
            <h3>Basic checkbox switch with custom labels</h3>
        </form>
    </div>
</div>

```

```

<input name="flip1" type="checkbox" data-role="flipswitch" data-
on-text="YES" data-off-text="NO" ><br/>

<input type="submit" value="Submit" name="submit" data-inline="true">
</form>
</div>
</div>
</body>
</html>

```

## Output

Let's carry out the following steps to see how the above code works:

- Save the above html code as **form\_flip\_toggle\_switch.html** file in your server root folder.
- Open this HTML file as [http://localhost/form\\_flip\\_toggle\\_switch.html](http://localhost/form_flip_toggle_switch.html) and the following output will be displayed.

## Form Multiple Selection

### Basic Flipswitch



### Basic checkbox switch with custom labels



**Submit**

## Refreshing and Auto Initialization of Form Elements

In this chapter, we will study about refreshing the form element and preventing auto-initialization of form elements.

### Refreshing Form Elements

Refresh method is used to update the new state of form control by itself and update the form control with JavaScript. Following code snippets are used for different form elements:

### Checkboxes

```

$( "input[type='checkbox']" ).prop( "checked",
true ).checkboxradio( "refresh" );

```

## Radios

```
$( "input[type='radio']" ).prop( "checked", true ).checkboxradio( "refresh" );
```

## Selects

```
var myselect = $( "#selectfoo" );
myselect[0].selectedIndex = 3;
myselect.selectmenu( "refresh" );
```

## Sliders

```
$( "input[type='range']" ).val( 60 ).slider( "refresh" );
```

## Flip switches

```
var myswitch = $( "#selectbar" );
myswitch[ 0 ].selectedIndex = 1;
myswitch.slider( "refresh" );
```

## Preventing Auto-initialization of Form Elements

Include the attribute **data-role="none"** to modify the selector which are used to prevent auto-initialization. Bound the **data-role="none"** attribute with **mobileinit** event to load the first page along with subsequent pages.

```
<label for="test">
<select id="test" name="test" data-role="none">
    <option value="Mumbai">Mumbai</option>
    <option value="Pune">Pune</option>
    <option value="Belgaum">Belgaum</option>
    <option value="Chennai">Chennai</option>
    <option value="Bangalore">Bangalore</option>
</select>
$( document ).bind( "mobileinit", function() {
    $.mobile.page.prototype.options.keepNative = "select, input.test,
    textarea.bar";
});
```

## 10. jQuery Mobile – Themes

It sets different types of theme on the buttons, navbars, blocks, links, and so on. You can set the theme using ***data-theme*** attribute.

Following table describes the use of the theme functionality in different areas, which is supported by jQuery Mobile.

| Sr. No. | Functionality & Description   |
|---------|---|
| 1       | <b><u>Themes</u></b><br>It provides two different types of themes such as theme "a" and theme "b" to customize the look of the application. |
| 2       | <b><u>Theming Header and Footer in Dialogs</u></b><br>Sets the theme for header and footer in the dialog box.                               |
| 3       | <b><u>Theming Buttons, Icons and Popups</u></b><br>Specifies the theme for buttons, icons, and popups.                                      |
| 4       | <b><u>Theming Buttons in Header and Footer</u></b><br>Displays the theme for buttons in the header and footer.                              |
| 5       | <b><u>Theming Navigation Bars</u></b><br>Applies the theme for navigation bars in header or footer.   |
| 6       | <b><u>Theming Panels</u></b><br>You can apply the theme for panel.  |
| 7       | <b><u>Theming Collapsible Button and Split Buttons</u></b><br>Displays the theme for collapsible and split buttons.                         |
| 8       | <b><u>Theming Lists and Collapsible Lists</u></b><br>Displays the theme for lists and collapsible lists.                                    |
| 9       | <b><u>Collapsible Forms</u></b><br>You can apply the theme for forms.   |

## Themes Types

---

### Description

You can set the different types of theme on the buttons, navbars, blocks, links, and so on. You can set the theme using ***data-theme*** attribute.

### Example

Following example demonstrates the use of themes in jQuery Mobile Framework.

```
<!DOCTYPE html>
<head>
<meta name="viewport" content="width=device-width, initial-scale=1">
<link rel="stylesheet"
href="https://code.jquery.com/mobile/1.4.5/jquery.mobile-1.4.5.min.css">
<script src="https://code.jquery.com/jquery-1.11.3.min.js"></script>
<script src="https://code.jquery.com/mobile/1.4.5/jquery.mobile-
1.4.5.min.js"></script>
</head>
<body>
<div data-role="page" id="pageone" data-theme="a">
<div data-role="panel" id="panel1">
<h2>This is Panel Header</h2>
<p>You can close this panel by pressing the Esc key or by swiping.</p>
</div>
<div data-role="header">
<h2>Header</h2>
</div>
<div data-role="main" class="ui-content">
<button class="ui-btn">Button</button>
<p>Click on the button to open Panel.</p>
<a href="#panel1" class="ui-btn ui-btn-inline">Open Panel</a>
<p>List View:</p>
<ul data-role="listview" data-autodividers="true">
<li><a href="#">India</a></li>
<li><a href="#">England</a></li>
<li><a href="#">Srilanka</a></li>
</ul><br>
```

```

<a href="#popup_menu" data-rel="popup" data-transition="slidedown"
class="ui-btn ui-btn-inline ui-icon-home ui-btn-icon-left ui-btn-a">Click to
see menu items</a>

<div data-role="popup" id="popup_menu" data-theme="a">
    <ul data-role="listview">
        <li data-role="list-divider">Select your choice</li>
        <li><a href="#">India</a></li>
        <li><a href="#">Srilanka</a></li>
        <li><a href="#">Australia</a></li>
        <li><a href="#">England</a></li>
    </ul>
</div>
</div>
<div data-role="footer">
    <h2>Footer</h2>
</div>
</div>
</body>
</html>

```

## Output

Let's carry out the following steps to see how the above code works:

- Save the above html code as **jqm\_themes\_types.html** file in your server root folder.
- Open this HTML file as [http://localhost/jqm\\_themes\\_types.html](http://localhost/jqm_themes_types.html) and the following output will be displayed.

## This is Panel Header

You can close this panel by pressing the Esc key or by swiping.

### Header

[Button](#)

Click on the button to open Panel.

[Open Panel](#)

List View:

- [India](#)
- [England](#)
- [Srilanka](#)

[Click to see menu items](#)

- Select your choice
- [India](#)
- [Srilanka](#)
- [Australia](#)
- [England](#)

### Footer

## Styling & Theming Dialogs

---

### Description

Different styling and theming can be applied to dialog. Following are the ways to apply theme and style to the dialog:

- Themes can be included in the dialog by adding attribute **data-theme** to any containers like header, footer, or content.
- Include **data-corners="false"** to remove the rounded corner of the dialog as by default the corners are set as rounded.
- **data-overlay-theme** attribute sets the overlay to any pattern letter.
- Multiple button can also be used in the dialogs. The dialog can be fixed to the top by including {margin-top:0} to the the class specified for the page **.ui-dialog.my-dialog .ui-dialog-contain**

## Example

Following example demonstrates the use of styling and theming dialogs in jQuery Mobile.

```
<!DOCTYPE html>
<html>
<head>
    <meta name="viewport" content="width=device-width, initial-scale=1">
    <link rel="stylesheet"
    href="https://code.jquery.com/mobile/1.4.5/jquery.mobile-1.4.5.min.css">
    <script src="https://code.jquery.com/jquery-1.11.3.min.js"></script>
    <script src="https://code.jquery.com/mobile/1.4.5/jquery.mobile-
1.4.5.min.js"></script>
    <style>
        .ui-dialog.dialog-actionsheet .ui-dialog-contain {
            margin-top: 0;
        }
    </style>
</head>
<body>
    <div data-role="page" id="page1">
        <div data-role="header" data-theme="b">
            <h2>Header</h2>
        </div>

        <div role="main" class="ui-content">
            <a href="#page2" class="ui-btn ui-corner-all ui-btn-inline">Open Dialog</a>
            <a href="#page3" class="ui-btn ui-corner-all ui-btn-inline">Photos Dialog</a>
        </div>

        <div data-role="footer">
            <h4>Footer</h4>
        </div>
    </div>

    <div data-role="page" data-dialog="true" id="page2" data-corners="false"
data-overlay-theme="b">
        <div data-role="header">
            <h1>First Dialogs</h1>
```

```

</div>

<div role="main" class="ui-content">
    <p>Lorem Ipsum is simply dummy text of the printing and typesetting industry. Lorem Ipsum has been the industry's standard dummy text ever since the 1500s.</p>
    <a data-rel="back" class="ui-btn ui-corner-all ui-btn-inline">Back</a>
</div>
</div>

<div data-role="page" class="dialog-actionsheet" data-dialog="true" id="page3">
    <div role="main" class="ui-content">
        <h3>Share Photos</h3>
        <a href="#page4" data-transition="slidedown" class="ui-btn ui-shadow ui-corner-all ui-btn-b">Email</a>
        <a href="#page4" data-transition="slidedown" class="ui-btn ui-shadow ui-corner-all ui-btn-b">Share on Facebook</a>
        <a href="#page4" data-transition="slidedown" class="ui-btn ui-shadow ui-corner-all ui-btn-b">Tweet photo</a>
        <a data-rel="back" class="ui-btn ui-shadow ui-corner-all ui-btn-a">Cancel</a>
    </div>
</div>

<div data-role="page" class="dialog-actionsheet" data-dialog="true" id="page4">
    <div role="main" class="ui-content">
        <h3>Photos uploaded successfully</h3>
        <a href="#page1" class="ui-btn ui-shadow ui-corner-all ui-btn-a">View photo page</a>
        <a href="#page1" class="ui-btn ui-shadow ui-corner-all ui-btn-a">Done</a>
    </div>
</div>
</body>
</html>

```

## Output

Let's carry out the following steps to see how the above code works:

- Save the above html code as **style\_theming.html** file in your server root folder.
- Open this HTML file as [http://localhost/style\\_theming.html](http://localhost/style_theming.html) and the following output will be displayed.

## Header

[Open Dialog](#) [Photos Dialog](#)

## Footer

# First Dialogs

Lorem Ipsum is simply dummy text of the printing and typesetting industry. Lorem Ipsum has been the industry's standard dummy text ever since the 1500s.

Back

## Share Photos

[Email](#) [Share on Facebook](#) [Tweet photo](#) [Cancel](#)

**Photos uploaded successfully**

[View photo page](#) [Done](#)

## Theming Buttons, Icons and Popups

---

### Description

Specify the theme for buttons, icons, and popups using the **ui-btn-theme\_name** class.

### Example

Following example demonstrates the use of theming buttons, icons, and popups in jQuery Mobile Framework.

```
<!DOCTYPE html>
<head>
<meta name="viewport" content="width=device-width, initial-scale=1">
<link rel="stylesheet"
href="https://code.jquery.com/mobile/1.4.5/jquery.mobile-1.4.5.min.css">
```

```

<script src="https://code.jquery.com/jquery-1.11.3.min.js"></script>
<script src="https://code.jquery.com/mobile/1.4.5/jquery.mobile-1.4.5.min.js"></script>
</head>
<body>
<div data-role="page" id="pageone" data-theme="a">
    <div data-role="panel" id="panel1">
        <h2>This is Panel Header</h2>
        <p>You can close this panel by pressing the Esc key or by swiping.</p>
    </div>
    <div data-role="header">
        <h2>Header</h2>
    </div>
    <div data-role="main" class="ui-content">
        <p>Buttons:</p>
        <a href="#" class="ui-btn ui-btn-inline"> Default Button</a>
        <a href="#" class="ui-btn ui-btn-inline ui-btn-b">Black Button</a>

        <p>Icons:</p>
        <a href="#" class="ui-btn ui-icon-home ui-btn-icon-notext">Home</a>
        <a href="#" class="ui-btn ui-btn-b ui-icon-home ui-btn-icon-notext">Home</a>

        <p>Popup:</p>
        <a href="#pop_demo" data-rel="popup" class="ui-btn ui-btn-b ui-btn-inline">Click to open popup box</a>
        <div id="pop_demo" data-role="popup" data-theme="a" class="ui-content">
            <a href="#" data-rel="back" class="ui-btn ui-btn-a ui-btn ui-icon-delete ui-btn-icon-notext ui-btn-right">Close</a>
            <p>Welcome to TutorialsPoint!!!</p>
        </div>
    </div>
    <div data-role="footer">
        <h2>Footer</h2>
    </div>
</div>
</body>
</html>

```

## Output

Let's carry out the following steps to see how the above code works:

- Save the above html code as **jqm\_theming\_buttons\_icons\_popups.html** file in your server root folder.
- Open this HTML file as [http://localhost/jqm\\_theming\\_buttons\\_icons\\_popups.html](http://localhost/jqm_theming_buttons_icons_popups.html) and the following output will be displayed.

## This is Panel Header

You can close this panel by pressing the Esc key or by swiping.

### Header

Buttons:

[Default Button](#) [Black Button](#)

Icons:

[Home](#) [Home](#)

Popup:

[Click to open popup box](#)  
[Close](#)

Welcome to TutorialsPoint!!!

### Footer

## Buttons in Footer

---

### Description

The **ui-btn-left** and **ui-btn-right** classes cannot be used in footer. However, you can still create the same appearance by using custom CSS.

### Example

Following example demonstrates the use of buttons in footers in jQuery Mobile.

```
<!DOCTYPE html>
<html>
<head>
```

```
<title>Buttons in footers</title>
<meta name="viewport" content="width=device-width, initial-scale=1">
<link rel="stylesheet"
href="https://code.jquery.com/mobile/1.4.5/jquery.mobile-1.4.5.min.css">
<script src="https://code.jquery.com/jquery-1.11.3.min.js"></script>
<script src="https://code.jquery.com/mobile/1.4.5/jquery.mobile-
1.4.5.min.js"></script>
<style>
    .footer-button-left,
    .footer-button-right {
        position: absolute;
        margin: 0;
        top: auto;
        bottom: 0.24em;
    }
    .footer-button-left {
        left: 0.4em;
    }
    .footer-button-right {
        right: 0.4em;
    }
</style>
</head>
<body>
    <div data-role="footer">
        <h2>Footer</h2>
        <a href="#" class="ui-btn ui-corner-all ui-btn-inline ui-mini footer-
button-left ui-btn-icon-left ui-icon-home">Home</a>
        <a href="#" class="ui-btn ui-corner-all ui-btn-inline ui-mini footer-
button-right ui-btn-icon-right ui-icon-gear">Options</a>
    </div>
</body>
</html>
```

## Output

Let's carry out the following steps to see how the above code works:

- Save the above html code as **toolbar\_buttons\_footer.html** file in your server root folder.
- Open this HTML file as [http://localhost/toolbar\\_buttons\\_footer.html](http://localhost/toolbar_buttons_footer.html) and the following output will be displayed.

## Footer

[Home](#)

[Options](#)

## Navbars Themes

---

### Description

Just like buttons, the theme swatch will be inherited by the navbars from their parent container. Whenever the navbar is positioned in the header or footer toolbar, the default toolbar swatch a will be inherited unless it is set in the markup.

You can add swatch classes to apply the body swatch (ui-body-a and ui-body-b). You can also use data-theme attribute to set theme color for individual navbar items and specify a theme swatch.

The class ui-body adds the standard body padding.

### Example

Following example demonstrates the use of navbar themes in jQuery Mobile.

```
<!DOCTYPE html>
<html>
<head>
    <title>Navbars Themes</title>
    <meta name="viewport" content="width=device-width, initial-scale=1">
    <link rel="stylesheet"
    href="https://code.jquery.com/mobile/1.4.5/jquery.mobile-1.4.5.min.css">
    <script src="https://code.jquery.com/jquery-1.11.3.min.js"></script>
```

```
<script src="https://code.jquery.com/mobile/1.4.5/jquery.mobile-1.4.5.min.js"></script>
</head>
<body>
<div class="ui-body-a ui-body">
    <h3>Swatch "a"</h3>
    <div data-role="navbar">
        <ul>
            <li><a href="#" data-icon="star">A</a></li>
            <li><a href="#" data-icon="gear">B</a></li>
            <li><a href="#" data-icon="grid">C</a></li>
            <li><a href="#" data-icon="arrow-l1">D</a></li>
            <li><a href="#" data-icon="arrow-r">E</a></li>
        </ul>
    </div>
</div>
<div class="ui-body-b ui-body">
    <h3>Swatch "b"</h3>
    <div data-role="navbar">
        <ul>
            <li><a href="#" data-icon="star">A</a></li>
            <li><a href="#" data-icon="gear">B</a></li>
            <li><a href="#" data-icon="grid">C</a></li>
            <li><a href="#" data-icon="arrow-l1">D</a></li>
            <li><a href="#" data-icon="arrow-r">E</a></li>
        </ul>
    </div>
</div>
<h3><i>data-theme</i> attribute</h3>
<div data-role="footer">
    <div data-role="navbar">
        <ul>
            <li><a href="#" data-icon="grid" data-theme="a">A</a></li>
            <li><a href="#" data-icon="grid" data-theme="b">B</a></li>
        </ul>
    </div>
</div>
</div>
```

```
</body>
</html>
```

## Output

Let's carry out the following steps to see how the above code works:

- Save the above html code as **navbar\_themes.html** file in your server root folder.
- Open this HTML file as [http://localhost/navbar\\_themes.html](http://localhost/navbar_themes.html) and the following output will be displayed.

### Swatch "a"

- [A](#)
- [B](#)
- [C](#)
- [D](#)
- [E](#)

### Swatch "b"

- [A](#)
- [B](#)
- [C](#)
- [D](#)
- [E](#)

### ***data-theme*** attribute

- [A](#)
- [B](#)

## Theming Panels

---

### Description

You can apply the theme for panel using the ***data-theme*** attribute.

### Example

Following example demonstrates the use of theming panels in jQuery Mobile Framework.

```
<!DOCTYPE html>
<head>
```

```

<meta name="viewport" content="width=device-width, initial-scale=1">
<link rel="stylesheet"
href="https://code.jquery.com/mobile/1.4.5/jquery.mobile-1.4.5.min.css">
<script src="https://code.jquery.com/jquery-1.11.3.min.js"></script>
<script src="https://code.jquery.com/mobile/1.4.5/jquery.mobile-
1.4.5.min.js"></script>
</head>
<body>
<div data-role="page" id="page1">
    <div data-role="panel" id="panel1" data-theme="b">
        <h2>This is Panel Header</h2>
        <p>You can close this panel by pressing the Esc key or by swiping.</p>
    </div>
    <div data-role="header">
        <h2>Header</h2>
    </div>
    <div data-role="main" class="ui-content">
        <p>Click on the button to open Panel.</p>
        <a href="#panel1" class="ui-btn ui-btn-inline">Open Panel</a>
    </div>
    <div data-role="footer">
        <h2>Footer</h2>
    </div>
</div>
</body>
</html>

```

## Output

Let's carry out the following steps to see how the above code works:

- Save the above html code as **jqm\_theming\_panels.html** file in your server root folder.
- Open this HTML file as [http://localhost/jqm\\_theming\\_panels.html](http://localhost/jqm_theming_panels.html) and the following output will be displayed.

## This is Panel Header

You can close this panel by pressing the Esc key or by swiping.

### Header

Click on the button to open Panel.

[Open Panel](#)

### Footer

## Theming Collapsible Button and Split Buttons

### Description

You can display the theme for collapsible and split buttons using the **data-theme** attribute.

### Example

Following example demonstrates the use of theming collapsible button and split buttons in jQuery Mobile Framework.

```
<!DOCTYPE html>
<head>
<meta name="viewport" content="width=device-width, initial-scale=1">
<link rel="stylesheet"
href="https://code.jquery.com/mobile/1.4.5/jquery.mobile-1.4.5.min.css">
<script src="https://code.jquery.com/jquery-1.11.3.min.js"></script>
<script src="https://code.jquery.com/mobile/1.4.5/jquery.mobile-
1.4.5.min.js"></script>
</head>
<body>
<div data-role="page" id="page1">
<div data-role="header">
    <h2>Header</h2>
</div>
<div data-role="main" class="ui-content">
    <p>Collapsible Button:</p>
```

```
<div data-role="collapsible" data-theme="a" data-content-theme="b">
    <h2>Collapsible Button</h2>
    <p>Welcome to Tutorialspoint!!!</p>
</div>
<p>Split Buttons:</p>
<ul data-role="listview" data-inset="true" data-split-theme="b">
    <li data-theme="a">Countries</li>
    <li>
        <a href="#">
            
            <h2>India</h2>
        </a>
        <a href="#my_img" data-rel="dialog" data-transition="pop"></a>
    </li>
    <li>
        <a href="#">
            
            <h2>Srilanka</h2>
        </a>
        <a href="#my_img" data-rel="dialog" data-transition="pop"></a>
    </li>
</ul>
</div>
<div data-role="footer">
    <h2>Footer</h2>
</div>
</div>
<div data-role="page" id="my_img">
    <div data-role="main" class="ui-content">
        <h2>Split Button</h2>
        <p>Do you want to download this logo?</p>
        <a href="#" class="ui-btn ui-btn-b ui-btn-inline ui-mini ui-icon-arrow-d ui-btn-icon-left" data-rel="back">Yes</a>
        <a href="#" class="ui-btn ui-btn-inline ui-mini ui-icon-delete ui-btn-icon-left" data-rel="back">No</a>
    </div>
</div>
```

```
</body>  
</html>
```

## Output

Let's carry out the following steps to see how the above code works:

- Save the above html code as **jqm\_theming\_collapsible\_split\_buttons.html** file in your server root folder.
- Open this HTML file as [http://localhost/jqm\\_theming\\_collapsible\\_split\\_buttons.html](http://localhost/jqm_theming_collapsible_split_buttons.html) and the following output will be displayed.

**Header**

Collapsible Button:

**Collapsible Button**

Welcome to TutorialsPoint!!!

Split Buttons:

- Countries



[\*\*India\*\*](#)



[\*\*Srilanka\*\*](#)

**Footer**

**Split Button**

Do you want to download this logo?

[Yes](#) [No](#)

## Theming Lists and Collapsible Lists

### Description

You can display the theme for collapsible and split buttons using the ***data-theme*** attribute.

### Example

Following example demonstrates the use of theming lists and collapsible lists in jQuery Mobile Framework.

```
<!DOCTYPE html>
<head>
<meta name="viewport" content="width=device-width, initial-scale=1">
<link rel="stylesheet"
href="https://code.jquery.com/mobile/1.4.5/jquery.mobile-1.4.5.min.css">
```

```
<script src="https://code.jquery.com/jquery-1.11.3.min.js"></script>
<script src="https://code.jquery.com/mobile/1.4.5/jquery.mobile-
1.4.5.min.js"></script>
</head>
<body>
<div data-role="page" id="page1">
    <div data-role="header">
        <h2>Header</h2>
    </div>
    <div data-role="main" class="ui-content">
        <p>List:</p>
        <ol data-role="listview" data-inset="true" data-theme="b">
            <li data-theme="a"><a href="#">India</a></li>
            <li data-theme="b"><a href="#">Srilanka</a></li>
            <li data-theme="a"><a href="#">Australia</a></li>
            <li data-theme="b"><a href="#">England</a></li>
        </ol>
        <br>
        <p>Collapsible List:</p>
        <div data-role="collapsible" data-theme="a" data-content-theme="b">
            <h2>Collapsible List</h2>
            <ul data-role="listview">
                <li><a href="#">India</a></li>
                <li><a href="#">Srilanka</a></li>
                <li><a href="#">Australia</a></li>
            </ul>
        </div>
    </div>
    <div data-role="footer">
        <h2>Footer</h2>
    </div>
</div>
</body>
</html>
```

## Output

Let's carry out the following steps to see how the above code works:

- Save the above html code as **jqm\_theming\_lists\_collapsible\_lists.html** file in your server root folder.
- Open this HTML file as [http://localhost/jqm\\_theming\\_lists\\_collapsible\\_lists.html](http://localhost/jqm_theming_lists_collapsible_lists.html) and the following output will be displayed.

## Header

List:

1. [India](#)
2. [Srilanka](#)
3. [Australia](#)
4. [England](#)

Collapsible List:

## Collapsible List

- [India](#)
- [Srilanka](#)
- [Australia](#)

## Footer

## Collapsible Forms

---

### Description

You can apply the theme for forms using the ***data-theme*** attribute within the form.

### Example

Following example demonstrates the use of collapsible forms in jQuery Mobile Framework.

```
<!DOCTYPE html>
<head>
<meta name="viewport" content="width=device-width, initial-scale=1">
<link rel="stylesheet"
href="https://code.jquery.com/mobile/1.4.5/jquery.mobile-1.4.5.min.css">
<script src="https://code.jquery.com/jquery-1.11.3.min.js"></script>
<script src="https://code.jquery.com/mobile/1.4.5/jquery.mobile-
1.4.5.min.js"></script>
</head>
<body>
<div data-role="page" id="page1">
    <div data-role="header">
        <h2>Header</h2>
    </div>
    <div data-role="main" class="ui-content">
        <form action="/jquery_mobile/submit.php" method="post">
            <fieldset data-role="collapsible" data-theme="b" data-content-theme="b">
                <legend>Collapsible Button</legend>
                <label for="name">First Name:</label>
                <input type="text" name="firstname" data-theme="a">
                <label for="name">Last Name:</label>
                <input type="text" name="lastname" data-theme="a">
                <input type="submit" data-inline="true" value="Submit" data-theme="a">
            </fieldset>
        </form>
    </div>
    <div data-role="footer">
        <h2>Footer</h2>
    </div>
</div>
```

```
</body>  
</html>
```

## Output

Let's carry out the following steps to see how the above code works:

- Save the above html code as **jqm\_collapsible\_forms.html** file in your server root folder.
- Open this HTML file as [http://localhost/jqm\\_collapsible\\_forms.html](http://localhost/jqm_collapsible_forms.html) and the following output will be displayed.

## Header

Collapsible Button

|                                       |                      |            |                      |
|---------------------------------------|----------------------|------------|----------------------|
| First Name:                           | <input type="text"/> | Last Name: | <input type="text"/> |
| <input type="button" value="Submit"/> |                      |            |                      |

## Footer

# 11. jQuery Mobile – CSS Classes

You can use different types of CSS classes to style the elements as described in the following sections.

## Global Classes

Following classes can be used as global classes on jQuery Mobile widgets.

| Sr. No. | Class & Description  |
|---------|--|
| 1       | <b>ui-corner-all</b><br>It displays the elements with rounded corners.       |
| 2       | <b>ui-shadow</b><br>It displays the shadow for the elements.                 |
| 3       | <b>ui-overlay-shadow</b><br>It displays the overlay shadow for the elements. |
| 4       | <b>ui-mini</b><br>It displays the smaller elements.                          |

## Button Classes

Following table lists button classes that are used with anchor or button elements.

| Sr. No. | Class & Description  |
|---------|--|
| 1       | <b>ui-btn</b><br>It specifies that the element will be styled as button.                                     |
| 2       | <b>ui-btn-inline</b><br>It shows the button as inline element which saves the space as needed for the label. |
| 3       | <b>ui-btn-icon-top</b><br>It places the icon above the text.   |
| 4       | <b>ui-btn-icon-right</b><br>It places the icon right of the text.  |

|   |   |
|---|---|
| 5 | <b>ui-btn-icon-bottom</b><br>It places the icon below the text.   |
| 6 | <b>ui-btn-icon-left</b><br>It places the icon left of the text.   |
| 7 | <b>ui-btn-icon-notext</b><br>It shows the only icon.  |
| 8 | <b>ui-btn-a b</b><br>It displays the color of the button ("a" will be the default background color, i.e. gray and "b" will change the background color to black). |

## Icon Classes

Following table lists icon classes that are used with anchor or button elements.

| Sr. No. | Class & Description  |
|---------|--|
| 1       | <b>ui-icon-action</b><br>It shows the action icon.                         |
| 2       | <b>ui-icon-alert</b><br>It display the exclamation mark inside a triangle. |
| 3       | <b>ui-icon-arrow-d-l</b><br>It specifies down with left arrow.             |
| 4       | <b>ui-icon-arrow-d-r</b><br>It specifies down with right arrow.            |
| 5       | <b>ui-icon-arrow-u-l</b><br>It specifies up with left arrow.               |
| 6       | <b>ui-icon-arrow-u-r</b><br>It specifies up with right arrow.              |
| 7       | <b>ui-icon-arrow-l</b><br>It specifies the left arrow.                     |
| 8       | <b>ui-icon-arrow-r</b><br>It specifies the right arrow.                    |
| 9       | <b>ui-icon-arrow-u</b><br>It specifies the up arrow.                       |

|    |  |
|----|--|
| 10 | <b>ui-icon-arrow-d</b><br>It specifies the down arrow.                           |
| 11 | <b>ui-icon-bars</b><br>It shows the 3 horizontal bars one above the other.       |
| 12 | <b>ui-icon-bullets</b><br>It shows the 3 horizontal bullets one above the other. |
| 13 | <b>ui-icon-carat-d</b><br>It displays the carat to down.                         |
| 14 | <b>ui-icon-carat-l</b><br>It displays the carat to left.                         |
| 15 | <b>ui-icon-carat-r</b><br>It displays the carat to right.                        |
| 16 | <b>ui-icon-carat-u</b><br>It displays the carat to up.                           |
| 17 | <b>ui-icon-check</b><br>It shows the checkmark icon.                             |
| 18 | <b>ui-icon-comment</b><br>It specifies the comment or message.                   |
| 19 | <b>ui-icon-forbidden</b><br>It displays the forbidden icon.                      |
| 20 | <b>ui-icon-forward</b><br>It specifies the forward icon.                         |
| 21 | <b>ui-icon-navigation</b><br>It specifies the navigation icon.                   |
| 22 | <b>ui-icon-recycle</b><br>It displays the recycle icon.                          |
| 23 | <b>ui-icon-refresh</b><br>It shows the refresh icon.                             |
| 24 | <b>ui-icon-tag</b><br>It indicates the tag icon.                                 |

|    |  |
|----|--|
| 25 | <b>ui-icon-video</b><br>It indicates the video or camera icon. |
|----|--|

## Theme Classes

---

It provides two different types of themes such as theme "a" and theme "b" to customize the look of the application. You can create your own theme classes by appending swatch letter (**a-z**). Following table lists theme classes, which are specified from letter **a** to **z**.

| Sr. No. | Class & Description  |
|---------|--|
| 1       | <b>ui-bar-(a-z)</b><br>It displays the color for bar including headers, footers, and other bars in the page.               |
| 2       | <b>ui-body-(a-z)</b><br>It displays the color for content block including listview, popups, sliders, panels, loaders, etc. |
| 3       | <b>ui-btn-(a-z)</b><br>It displays the color for button.   |
| 4       | <b>ui-group-theme-(a-z)</b><br>It displays the color for controlgroups, listviews, and collapsible sets.                   |
| 5       | <b>ui-overlay-(a-z)</b><br>It displays the background color for popup, dialog, and page containers.                        |
| 6       | <b>ui-page-theme-(a-z)</b><br>It displays the color for pages.   |

## Grid Classes

---

Following table lists grid classes that are used with equal width, no border, background, margin or padding.

| Sr. No. | Grid Class   | Columns | Column Widths       | Corresponds To     |
|---------|--------------|---------|---------------------|--------------------|
| 1       | ui-grid-solo | 1       | 100%                | ui-block-a         |
| 2       | ui-grid-a    | 2       | 50%/50%             | ui-block-a b       |
| 3       | ui-grid-b    | 3       | 33%/33%/33%         | ui-block-a b c     |
| 4       | ui-grid-c    | 4       | 25%/25%/25%/25%     | ui-block-a b c d   |
| 5       | ui-grid-d    | 5       | 20%/20%/20%/20%/20% | ui-block-a b c d e |

## 12. jQuery Mobile – Data Attributes

### Button

It specifies a clickable button that includes content like text or images using the class **ui-btn**. It is deprecated in version 1.4. Use the **ui-btn** attribute instead of using **data-role="button"** attribute.

Following table lists the button elements used with data attribute.

| Sr. No. | Data-attribute & Description  | Value                       |
|---------|---|-----------------------------|
| 1       | <b>data-corners</b><br>It defines whether the button should contain rounded corners or not.       | true   false                |
| 2       | <b>data-icon</b><br>It defines the icon of the button.  | Default is no icon          |
| 3       | <b>data-iconpos</b><br>It defines the position of the icon.                                       | left   right   top   bottom |
| 4       | <b>data-iconshadow</b><br>It defines whether the icon of the button should contain shadow or not. | true   false                |
| 5       | <b>data-inline</b><br>It defines whether the button should be inline or not.                      | true   false                |
| 6       | <b>data-mini</b><br>It defines whether the button should display in smaller size or regular size. | true   false                |
| 7       | <b>data-shadow</b><br>It defines whether the button should contain shadow or not.                 | true   false                |
| 8       | <b>data-theme</b><br>It displays the theme color for the button.                                  | letter (a-z)                |

## Checkbox

---

Following table lists the checkbox elements used with `type="checkbox"`.

| Sr. No. | Data-attribute & Description  | Value        |
|---------|---|--------------|
| 1       | <b>data-mini</b><br>It defines whether the checkbox should display in smaller size or regular size. | true   false |
| 2       | <b>data-role</b><br>It stops styling of checkboxes as buttons.                                      | none         |
| 3       | <b>data-theme</b><br>It displays the theme color for the checkbox.                                  | letter (a-z) |

## Collapsible

---

Following table lists collapsible elements used with `data-role="collapsible"` attribute.

| Sr. No. | Data-attribute & Description  | Value                         |
|---------|---|-------------------------------|
| 1       | <b>data-collapsed</b><br>It indicates whether the content should be closed or expanded.       | true   false                  |
| 2       | <b>data-collapsed-cue-text</b><br>It displays feedback for users with screen reader software. | Default is collapsing content |
| 3       | <b>data-collapsed-icon</b><br>It defines the icon of collapsible button.                      | Default icon is "plus"        |
| 4       | <b>data-content-theme</b><br>It displays the theme color for the collapsible content.         | letter (a-z)                  |
| 5       | <b>data-expanded-cue-text</b><br>It displays feedback for users with screen reader software.  | Default is expanding content  |
| 6       | <b>data-expanded-icon</b><br>It displays the collapsible button when you expand the content.  | Default icon is "minus"       |

|    |   |                             |
|----|---|-----------------------------|
| 7  | <b>data-iconpos</b><br>It defines the position of the icon.   | left   right   top   bottom |
| 8  | <b>data-inset</b><br>It defines whether the collapsible button should display with rounded corners and margin or not. | true   false                |
| 9  | <b>data-mini</b><br>It defines whether the collapsible buttons should display in smaller size or regular size.        | true   false                |
| 10 | <b>data-theme</b><br>It displays the theme color for the collapsible button.  | letter (a-z)                |

## Collapsible Set

Following table lists the collapsible set elements used with the **data-role="collapsibleset"** attribute.

| Sr. No. | Data-attribute & Description  | Value                       |
|---------|---|-----------------------------|
| 1       | <b>data-collapsed-icon</b><br>It defines the icon of collapsible button.  | Default icon is "plus"      |
| 2       | <b>data-content-theme</b><br>It displays the theme color for the collapsible content.                                 | letter (a-z)                |
| 3       | <b>data-expanded-icon</b><br>It displays the collapsible button when you expand the content.                          | Default icon is "minus"     |
| 4       | <b>data-iconpos</b><br>It defines the position of the icon.   | left   right   top   bottom |
| 5       | <b>data-inset</b><br>It defines whether the collapsible button should display with rounded corners and margin or not. | true   false                |
| 6       | <b>data-mini</b><br>It defines whether the collapsible buttons should display in smaller size or regular size.        | true   false                |
| 7       | <b>data-theme</b><br>It displays the theme color for the collapsible button.  | letter (a-z)                |

## Controlgroup

---

Following table lists Controlgroup elements used with ***data-role="controlgroup"*** attribute.

| Sr. No. | Data-attribute & Description   | Value                 |
|---------|--|-----------------------|
| 1       | <b><i>data-exclude-invisible</i></b><br>It defines whether to exclude invisible children in the assignment of rounded corners. | true   false          |
| 2       | <b><i>data-mini</i></b><br>It defines whether the group should display in smaller size or regular size.                        | true   false          |
| 3       | <b><i>data-theme</i></b><br>It displays the theme color for the controlgroup.  | letter (a-z)          |
| 4       | <b><i>data-type</i></b><br>It indicates whether the group should display in horizontal or vertical format.                     | horizontal   vertical |

## Dialog

---

Following table lists dialog elements used with ***data-dialog="true"*** attribute.

| Sr. No. | Data-attribute & Description   | Value               |
|---------|--|---------------------|
| 1       | <b><i>data-close-btn</i></b><br>It defines the position of the close button.                           | left   right   none |
| 2       | <b><i>data-close-btn-text</i></b><br>It defines the text for the close button.                         | text                |
| 3       | <b><i>data-corners</i></b><br>It defines whether dialog should display with rounded corners or not.    | true   false        |
| 4       | <b><i>data-dom-cache</i></b><br>It indicates whether DOM cache must clear or not for individual pages. | true   false        |
| 5       | <b><i>data-overlay-theme</i></b><br>It defines the overlay color of the dialog page.                   | letter (a-z)        |

|   |   |              |
|---|---|--------------|
| 6 | <b>data-theme</b><br>It defines the theme color of the dialog page. | letter (a-z) |
| 7 | <b>data-title</b><br>It defines the title of the dialog page.       | text         |

## Enhancement

---

Following table lists enhancement elements used with **data-enhance="false"** or **data-ajax="false"** attribute.

| Sr. No. | Data-attribute & Description  | Value        |
|---------|---|--------------|
| 1       | <b>data-enhance</b><br>You can style the page by setting this attribute to "true". You cannot style the page if it is set to "false". | true   false |
| 2       | <b>data-ajax</b><br>It indicates whether pages must load from Ajax or not.  | true   false |

## Fixed Toolbar

---

Following table lists the toolbar elements used with **data-position="fixed"** attribute.

| Sr. No. | Data-attribute & Description  | Value               |
|---------|---|---------------------|
| 1       | <b>data-disable-page-zoom</b><br>It defines whether the user is able to scale/zoom the page or not.   | true   false        |
| 2       | <b>data-fullscreen</b><br>It defines toolbars must be positioned at the top and/or bottom.            | true   false        |
| 3       | <b>data-tap-toggle</b><br>It indicates whether the user can toggle toolbar-visibility on taps or not. | true   false        |
| 4       | <b>data-transition</b><br>It shows a transition effect when you tap or click the element.             | slide   fade   none |

|   |  |              |
|---|--|--------------|
| 5 | <b>data-update-page-padding</b><br>It updates the padding of page by using resize, transition, and update layout events. | true   false |
| 6 | <b>data-visible-on-page-show</b><br>It defines toolbar-visibility when the parent page is shown.                         | true   false |

## Flip Toggle Switch

Following table lists the flip toggle elements used with **data-role="flipswitch"** attribute.

| Sr. No. | Data-attribute & Description  | Value            |
|---------|---|------------------|
| 1       | <b>data-mini</b><br>It defines whether the switch should display in smaller size or regular size. | true   false     |
| 2       | <b>data-on-text</b><br>It defines the "on" text on the flip switch.                               | Default is "on"  |
| 3       | <b>data-off-text</b><br>It defines the "off" text on the flip switch.                             | Default is "off" |

## Footer

Following table lists the footer elements used with **data-role="footer"** attribute.

| Sr. No. | Data-attribute & Description   | Value          |
|---------|--|----------------|
| 1       | <b>data-id</b><br>It defines the unique ID.  | text           |
| 2       | <b>data-position</b><br>It defines whether the footer should be positioned at the bottom or inline with page content.        | inline   fixed |
| 3       | <b>data-fullscreen</b><br>It defines whether the footer should be positioned at the bottom and over the page content or not. | true   false   |
| 4       | <b>data-theme</b><br>It defines the theme color of the footer.   | letter (a-z)   |

## Header

---

Following table lists the header elements used with ***data-role="header"*** attribute.

| Sr. No. | Data-attribute & Description  | Value          |
|---------|---|----------------|
| 1       | <b><i>data-id</i></b><br>It defines the unique ID.  | text           |
| 2       | <b><i>data-position</i></b><br>It defines whether the header should be positioned at the bottom or inline with the page content.    | inline   fixed |
| 3       | <b><i>data-fullscreen</i></b><br>It defines whether the header should be positioned at the bottom and over the page content or not. | true   false   |
| 4       | <b><i>data-theme</i></b><br>It defines the theme color of the header.   | letter (a-z)   |

## Inputs

---

Following table lists the input elements used with ***type="text|search|etc"*** attribute.

| Sr. No. | Data-attribute & Description   | Value        |
|---------|--|--------------|
| 1       | <b><i>data-clear-btn</i></b><br>It defines whether the input element should contain clear button or not. | true   false |
| 2       | <b><i>data-clear-btn-text</i></b><br>It defines the text for the clear button.                           | text         |
| 3       | <b><i>data-mini</i></b><br>It defines whether the input should display in smaller size or regular size.  | true   false |
| 4       | <b><i>data-role</i></b><br>It stops styling input or text areas as buttons.                              | none         |
| 5       | <b><i>data-theme</i></b><br>It defines the theme color of the input element.                             | letter (a-z) |

## Link

---

Following table lists the link elements used with jQuery Mobile.

| Sr. No. | Data-attribute & Description   | Value  |
|---------|--|--|
| 1       | <b>data-ajax</b><br>It indicates whether the pages must be loaded through Ajax or not.         | true   false   |
| 2       | <b>data-direction</b><br>It is used for reverse transition.                                    | reverse  |
| 3       | <b>data-dom-cache</b><br>It indicates whether jQuery DOM cache must be clear or not for pages. | true   false   |
| 4       | <b>data-prefetch</b><br>It is used to prefetch the pages into DOM.                             | true   false   |
| 5       | <b>data-rel</b><br>It specifies the behavior of the link.                                      | back   dialog   external   popup   |
| 6       | <b>data-transition</b><br>It defines the transition from one page to another.                  | fade   flip   flow   pop   slide   slidedown   slidefade   slideup   turn   none |
| 7       | <b>data-position-to</b><br>It defines the position of the popup boxes.                         | origin   jQuery selector   window  |

## List

---

Following table shows the list elements used with **data-role="listview"** attribute.

| Sr. No. | Data-attribute & Description  | Value        |
|---------|---|--------------|
| 1       | <b>data-autodividers</b><br>It divides the list automatically.              | true   false |
| 2       | <b>data-count-theme</b><br>It defines the theme color of the count element. | letter (a-z) |
| 3       | <b>data-divider-theme</b><br>It defines the theme color for list divider.   | letter (a-z) |

|    |   |                               |
|----|---|-------------------------------|
| 4  | <b>data-filter</b><br>It is used to filter the list values in the search box.                       | true   false                  |
| 5  | <b>data-filter-placeholder</b><br>It defines some text inside the search box.                       | text                          |
| 6  | <b>data-filter-theme</b><br>It defines the theme color for search filter.                           | letter (a-z)                  |
| 7  | <b>data-icon</b><br>It provides the icon for the list.  | Default is no icon            |
| 8  | <b>data-inset</b><br>It defines whether list should display with rounded corners and margin or not. | true   false                  |
| 9  | <b>data-split-icon</b><br>It defines the icon for split button.                                     | The default icon is "arrow-r" |
| 10 | <b>data-split-theme</b><br>It defines the theme color for split button.                             | letter (a-z)                  |
| 11 | <b>data-theme</b><br>It defines the theme color for the list.                                       | letter (a-z)                  |

## List Item

Following table shows the list item elements used with **data-role="listview"** attribute.

| Sr. No. | Data-attribute & Description   | Value              |
|---------|--|--------------------|
| 1       | <b>data-filtertext</b><br>It is used to filter the list values using the text in the search box. | text               |
| 2       | <b>data-icon</b><br>It provides the icon for the list item.                                      | Default is no icon |
| 3       | <b>data-role</b><br>It defines the divider for list items.                                       | list-divider       |
| 4       | <b>data-theme</b><br>It defines the theme color for the list item.                               | letter (a-z)       |

## Navbar

---

Following table lists navbar elements used with ***data-role="navbar"*** attribute.

| Sr. No. | Data-attribute & Description                                 | Value                                |
|---------|--|--------------------------------------|
| 1       | <b>data-icon</b><br>It provides the icon for the list item.  | Default is no icon                   |
| 2       | <b>data-iconpos</b><br>It defines the position for the icon. | left   right   top   bottom   notext |

## Page

---

Following table lists the page elements used with ***data-role="page"*** attribute.

| Sr. No. | Data-attribute & Description  | Value              |
|---------|---|--------------------|
| 1       | <b>data-dom-cache</b><br>It indicates whether DOM cache must clear or not for individual pages. | true   false       |
| 2       | <b>data-overlay-theme</b><br>It defines the overlay color of the dialog pages.                  | letter (a-z)       |
| 3       | <b>data-theme</b><br>It defines theme color for the page.                                       | letter (a-z)       |
| 4       | <b>data-title</b><br>It provides the title for the page.  | Default is no icon |
| 5       | <b>data-url</b><br>It is used to updating the URL.  | url                |

## Popup

---

Following table lists the popup elements used with ***data-role="popup"*** attribute.

| Sr. No. | Data-attribute & Description   | Value        |
|---------|--|--------------|
| 1       | <b>data-corners</b><br>It defines whether the popup should display with rounded corners and margin or not. | true   false |

|   |   |                |
|---|---|----------------|
| 2 | <b>data-dismissible</b><br>It defines whether the popup should be close by clicking outside or not. | true   false   |
| 3 | <b>data-history</b><br>It defines whether the popup should display the history of item when opened. | true   false   |
| 4 | <b>data-overlay-theme</b><br>It defines the overlay color of the popup box.                         | letter (a-z)   |
| 5 | <b>data-shadow</b><br>It displays the shadow for the popup box.                                     | true   false   |
| 6 | <b>data-theme</b><br>It defines the theme color for the popup box.                                  | letter (a-z)   |
| 7 | <b>data-tolerance</b><br>It defines the edges of the window.  | 30, 15, 30, 15 |

## Radio Button

---

Following table lists the radio button elements used with **`type="radio"`** attribute.

| Sr. No. | Data-attribute & Description  | Value        |
|---------|---|--------------|
| 1       | <b>data-mini</b><br>It defines whether the button should display in smaller size or regular size. | true   false |
| 2       | <b>data-role</b><br>It stops the styling of radio buttons as enhanced buttons.                    | none         |
| 3       | <b>data-theme</b><br>It defines the theme color for the radio button.                             | letter (a-z) |

## Select

---

Following table lists the select elements used with jQuery Mobile.

| Sr. No. | Data-attribute & Description  | Value                       |
|---------|---|-----------------------------|
| 1       | <b>data-icon</b><br>It provides the icon for the select element.                              | Default is "arrow-d"        |
| 2       | <b>data-iconpos</b><br>It defines the position of the icon.                                   | left   right   top   bottom |
| 3       | <b>data-inline</b><br>It defines whether the button should be inline or not.                  | true   false                |
| 4       | <b>data-mini</b><br>It defines whether select should display in smaller size or regular size. | true   false                |
| 5       | <b>data-native-menu</b><br>It use custom menu when it has been set to false.                  | true   false                |
| 6       | <b>data-overlay-theme</b><br>It defines the overlay color for the custom select menu.         | letter (a-z)                |
| 7       | <b>data-placeholder</b><br>It is used to set an option element of non-native select.          | true   false                |
| 8       | <b>data-role</b><br>It stops the styling of select elements as buttons.                       | none                        |
| 9       | <b>data-theme</b><br>It displays the theme color for the select elements.                     | letter (a-z)                |

## Slider

---

Following table lists the slider elements used with ***type="range"*** attribute.

| Sr. No. | Data-attribute & Description  | Value        |
|---------|---|--------------|
| 1       | <b>data-highlight</b><br>It highlights the slider.  | true   false |
| 2       | <b>data-mini</b><br>It defines whether the slider should display in smaller size or regular size. | true   false |
| 3       | <b>data-role</b><br>It stops the styling of slider controls as buttons.                           | none         |
| 4       | <b>data-theme</b><br>It displays the theme color for the slider control.                          | letter (a-z) |
| 5       | <b>data-track-theme</b><br>It displays the theme color for the slider track.                      | letter (a-z) |