

# Getting Started

Qmethod



# Edit startingPages.xml file located at settings folder to customize your starting pages

```
<?xml version="1.0" encoding="UTF-8"?>
<pages>
  <div>
    <p style="margin-left:100px; margin-right:100px; margin-bottom:20px; margin-top:50px;">
      Page 1: Example
    </p>
  </div>
  <div>
    <p style="margin-left:100px; margin-right:100px; margin-bottom:20px; margin-top:50px;">
      Page 2: Add how many pages you wish
    </p>
  </div>
  <div>
    <p style="margin-left:100px; margin-right:100px; margin-bottom:20px; margin-top:50px;">
      Page 3: by editing startingPages.xml file located at settings folder.
    </p>
  </div>
  <div>
    <h3>Page 4: With whatever formatting you wish, by using html tags.</h3>
  </div>
</pages>
```

More on the next page...

```

<?xml version="1.0" encoding="UTF-8"?>
<p>pages</p>
  <div>
    <p style="...">Page 1</p>
  </div>
  <p style="...">
    Page 2
  </p>
  <div>
    <h3>Page 3</h3>
  </div>
</pages>

```

Each node is a page. Here, `<div>` is and everything contained within is a page.

```

<?xml version="1.0" encoding="UTF-8"?>
<pages>
  <div>
    <p style="...">Page 1</p>
  </div>
  <p style="...">
    Pag 2
  </p>
  <div>
    <h3>Page 3</h3>
  </div>
</pages>

```

Here, `<p>` and everything contained within is a page.

Other html tags such as `<h3>` might be used as well.

```
<?xml version="1.0" encoding="UTF-8"?>
<pages>
  <div>
    <p style="margin-left:100px; margin-right:100px; margin-bottom:20px; margin-top:50px;">
      Page 1: Example
    </p>
  </div>
  <div>
    <p style="margin-left:100px; margin-right:100px; margin-bottom:20px; margin-top:50px;">
      Page 2: Add how many pages you wish
    </p>
  </div>
  <div>
    <p style="margin-left:100px; margin-right:100px; margin-bottom:20px; margin-top:50px;">
      Page 3: by editing startingPages.xml file located at settings folder.
    </p>
  </div>
  <div>
    <h3>Page 4: With whatever formatting you wish, by using html tags.</h3>
  </div>
</pages>
```

Styling defined on `style="..."` attribute applies as well.

Anything HTML is valid! You can find more content about HTML at : <https://www.w3schools.com/html/>

# Pages generated from previous XML example

<p>Page 1: Example</p> <p>Continue</p>	<p>Page 4: With whatever formatting you wish, by using html tags.</p> <p>Back Continue</p>
<p>Page 2: Add how many pages you wish</p> <p>Back Continue</p>	<p>Page 3: by editing startingPages.xml file located at settings folder.</p> <p>Back Continue</p>

# Edit statements.xml file at settings folder to customize your questions

```
<?xml version="1.0" encoding="UTF-8"?>
<statements>
  <statement id="1">Statement example 1.</statement>
  <statement id="2">Statement example 2.</statement>
  <statement id="3">Statement example 3.</statement>
  <statement id="4">Statement example 4.</statement>
  <statement id="5">Statement example 5.</statement>
</statements>
```

**This xml file defines the statements cards.**

id : question/statement identification number.

Statements cards generated by previous example xml; Some have been distributed for this example.

If you need to read the instructions of this step again, click on the "Help" button on the bottom this page.

Statement example 1.

DISAGREE

Statement example 2.

NEUTRAL

Statement example 4.

AGREE

Statement example 3.

Statement example 5.

Back

Help

The model for this figure would be:

```
"classifications": {"AGREE": ["2"],"NEUTRAL":["4"], "DISAGREE":["3","5"]}
```

# Edit map.xml file at settings folder to customize the normal distribution

```
<?xml version="1.0" encoding="UTF-8"?>
<map>
  <column id="3" colour="#06960D">2</column>
  <column id="2" colour="#45B149">6</column>
  <column id="1" colour="#5ABA5D">10</column>
  <column id="0" colour="#969292">14</column>
  <column id="-1" colour="#FF8989">10</column>
  <column id="-2" colour="#FF7171">6</column>
  <column id="-3" colour="#E82A2A">2</column>
</map>
```

**This xml file defines the normal distribution.**

id : column header label;

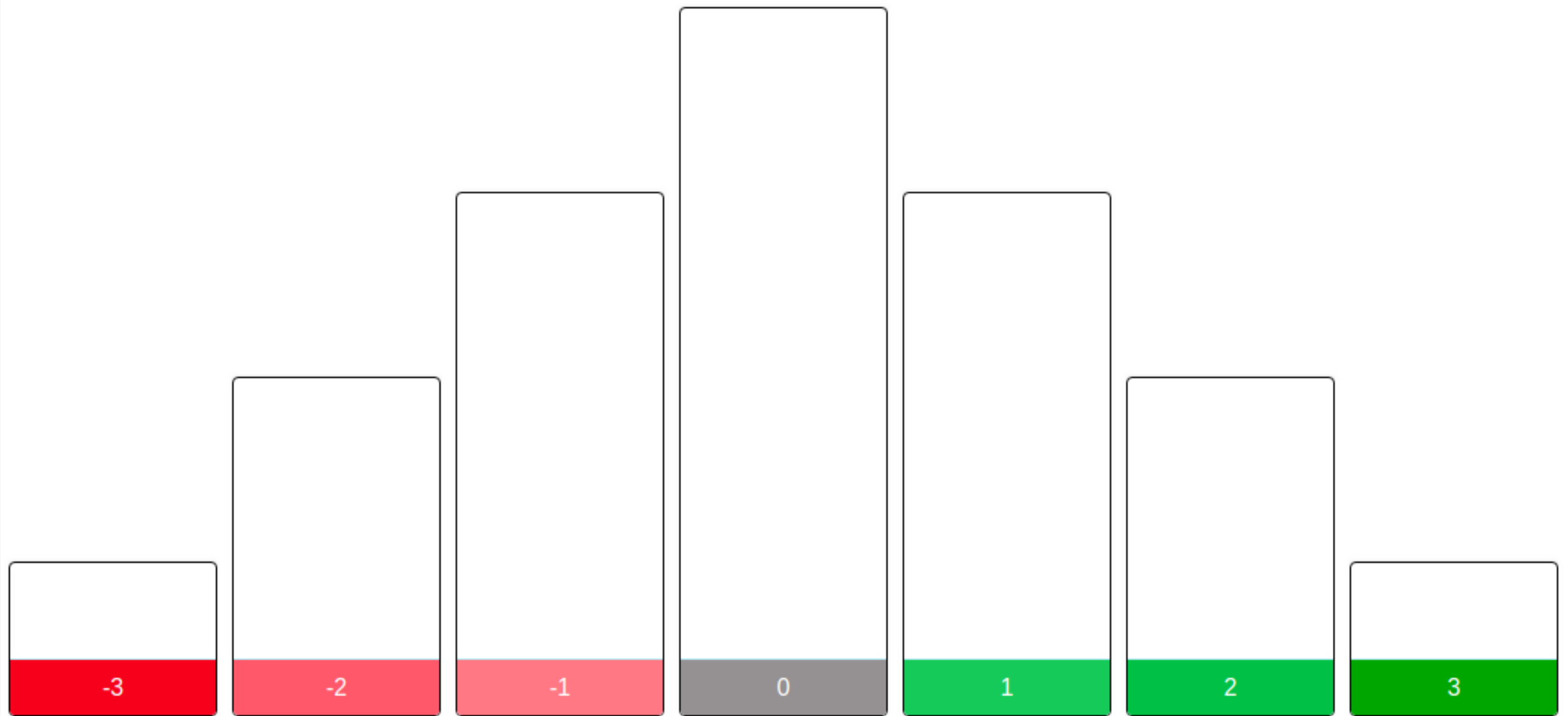
colour: column header color;

value: how many rows the column might hold.

Note that the sum of all values  $(2+6+10)*2 + 14$  must be equals to the number of statements you have configured, in this case, we should have 50 statements.



If you need to read the instructions of this step again, click on the "Help" button on the bottom fo this page.



**Normal distribution generated by previous xml.**

# Edit settings/survey.xml to customize the final survey

```
<?xml version="1.0" encoding="UTF-8"?>
<map>
  <input type="input" required="true" key="age">
    <label>Please enter your year of birth (YYYY, eg. 1980).</label>
  </input>
  <input type="radio" required="true" key="gender">
    <label>Please select your gender.</label>
    <option name="female">female</option>
    <option name="male">male</option>
  </input>
  <input type="select" required="true" key="edu">
    <label>Please select your educational level.</label>
    <option name="PhD">phd</option>
    <option name="PhD Candidate">phdc</option>
    <option name="Master">msc</option>
  </input>
</map>
```

**This xml configures the final survey.**

More documentation on next page...

## Input attributes:

- type : the type of input. Supported are select, input, radio and textarea;
- required: whether the field is required or not. May be assigned as true or false;
- key: key in which the data will be stored.

Radio and select have additional 'option' tags, which defines the options that will be available to be selected.

## Option attributes:

- name: how option will be shown;
- value (inside tag): how option will be stored.

Please enter your year of birth (YYYY, eg. 1980). \*

Please select your gender. \*

☐ female

☐ male

Please select your educational level. \*


Survey generated by previous xml.

You might edit html files contained at templates folder for more customization.

# Configuring Firebase



Go to [firebase.com](https://firebase.com) > create account/login > Add Project;

After creating your project, to get the configuration, click on '</>' button or click on the gear  > Your apps > Add Firebase to your web app > copy everything that is inside var config = {... ;

Open the file qmethod.js located at src folder, then find the line with var = config...

```
var config = {  
};
```

Paste your Firebase settings into it.

```
var config = {  
  apiKey: "your api key",  
  authDomain: "your auth domain",  
  databaseURL: "your db url",  
  projectId: "your project id",  
  storageBucket: "your store bucket",  
  messagingSenderId: "your msg sender id"  
};
```

Finally, click at Develop > Database > Create database > start in test mode > enable

**Warning!** : test mode will start with rules that **allows anyone to read from and write to your database**. For more information on how to configure rules, head to : <https://firebase.google.com/docs/database/security/quickstart>



# Deploying/Running locally

UNIX-like (MacOs, GNU/Linux, BSD etc)

Have python3 installed; cd into qmethod, then:

```
python3 -m http.server 8080 --bind 127.0.0.1
```

Then, type in your browser address bar:  
127.0.0.1:8080

Windows 7/8/10

Similar to previous or, install Fenix Web Server:  
<http://fenixwebserver.com/>

# Deploying/Running on internet

Use your favorite web hosting service.