

Name: Goudamy Sockalingam

Student ID: 009417148

## Registration Page

### Part 1: Create a registration page:

The basic registration page:

#### Registration Page

First Name:	<input type="text" value="Enter First Name"/>
Last Name:	<input type="text" value="Enter Last Name"/>
email:	<input type="text" value="Enter email-id"/>
password:	<input type="password" value="Enter password"/> Password must contain at least three uppercase letters, at least two lowercase letters, and two special characters.
confirm password:	<input type="password" value="Must be same as password"/>
D.O.B	<input type="text" value="mm/dd/yyyy"/>
D.O.B with time	<input type="text"/>
Local D.O.B	<input type="text" value="mm/dd/yyyy --:-- --"/>
SSN	<input type="text"/>
Phone	<input type="text"/>
Credit card number	<input type="text"/>
<input type="button" value="create"/>	

When the user provides password, it would check the password strength on the fly.

#### Registration Page

First Name:	<input type="text" value="Goudamy"/>
Last Name:	<input type="text" value="Sockalingam"/>
email:	<input type="text" value="goudamy.sg@gmail.com"/>
password:	<input type="password" value="****"/> Not secure
confirm password:	<input type="password" value="Must be same as password"/>
D.O.B	<input type="text" value="mm/dd/yyyy"/>
D.O.B with time	<input type="text"/>
Local D.O.B	<input type="text" value="mm/dd/yyyy --:-- --"/>
SSN	<input type="text"/>
Phone	<input type="text"/>
Credit card number	<input type="text"/>
<input type="button" value="create"/>	

The progress bar would depict the strength as below:

- If password contains 3 Uppercase letters, 2 Lowercase letters and 2 special characters, then Password strength is **Normal**
- If password contains 3 Uppercase letters, 2 Lowercase letters and **3-5** special characters, then Password strength is **Medium**

- If password contains 3 Uppercase letters, 2 Lowercase letters and ***greater than 5*** special characters, then Password strength is **Strong**

### Registration Page

First Name:	<input type="text" value="Goudamy"/>	
Last Name:	<input type="text" value="Sockalingam"/>	
email:	<input type="text" value="goudamy.s@gmail.com"/>	
password:	<input type="password" value="*****"/>	<div><div></div></div> Normal!! But its easy to predict
confirm password:	<input type="text" value="Must be same as password"/>	
D.O.B	<input type="text" value="mm/dd/yyyy"/>	
D.O.B with time	<input type="text"/>	
Local D.O.B	<input type="text" value="mm/dd/yyyy --:-- --"/>	
SSN	<input type="text"/>	
Phone	<input type="text"/>	
Credit card number	<input type="text"/>	
<input type="button" value="create"/>		

### Registration Page

First Name:	<input type="text" value="Goudamy"/>	
Last Name:	<input type="text" value="Sockalingam"/>	
email:	<input type="text" value="goudamy.s@gmail.com"/>	
password:	<input type="password" value="*****"/>	<div><div></div></div> Good!! But it can be better
confirm password:	<input type="text" value="Must be same as password"/>	
D.O.B	<input type="text" value="mm/dd/yyyy"/>	
D.O.B with time	<input type="text"/>	
Local D.O.B	<input type="text" value="mm/dd/yyyy --:-- --"/>	
SSN	<input type="text"/>	
Phone	<input type="text"/>	
Credit card number	<input type="text"/>	
<input type="button" value="create"/>		

## Registration Page

Registration Page

First Name:

Last Name:

email:

password:  Strong

confirm password:

D.O.B

D.O.B with time

Local D.O.B

SSN

Phone

Credit card number

If the password and confirm password doesn't match , then an alert window is used to display the message

localhost:8383/HTML5Application/Registrartion.html?fname=&lname=&email=&password=ram&password1=ra&bday=&bday1=&bday2=&ssn=&phone=&cc=

The page at localhost:8383 says:

Passwords dont match

OK

First Name:

Last Name:

email:

password:  Strong

confirm password:

D.O.B

D.O.B with time

Local D.O.B

SSN

Phone

Credit card number

## Part 2: Using HTML5 create pattern for

- SSN
- US phone number
- Credit card number (12 digit / 16 digit)

Pattern(XXX-XX-XXXX) is created for the SSN. If the pattern is not followed, it would display a message as below.

### Registration Page

First Name:	<input type="text" value="Goudamy"/>	
Last Name:	<input type="text" value="Sockalingam"/>	
email:	<input type="text" value="goudamy.s@gmail.com"/>	
password:	<input type="password" value="*****"/>	<div><div></div><div>Strong</div></div>
confirm password:	<input type="password" value="*****"/>	
D.O.B	<input type="text" value="04/01/1989"/>	
D.O.B with time	<input type="text" value="12 PM"/>	
Local D.O.B	<input type="text" value="04/01/1989 05:15 PM"/>	
SSN	<input type="text" value="510-45-12346"/>	
Phone	<input type="text" value="510-857-4"/>	
Credit card number	<input type="text" value="1234-4567"/>	

Please match the requested format.  
XXX-XX-XXXX

create

Pattern(XXX-XXX-XXXX) is created for the Phone number. If the pattern is not followed, it would display a message as below.

### Registration Page

First Name:	<input type="text" value="Goudamy"/>	
Last Name:	<input type="text" value="Sockalingam"/>	
email:	<input type="text" value="goudamy.s@gmail.com"/>	
password:	<input type="password" value="*****"/>	<div><div></div><div>Strong</div></div>
confirm password:	<input type="password" value="*****"/>	
D.O.B	<input type="text" value="04/01/1989"/>	
D.O.B with time	<input type="text" value="12 PM"/>	
Local D.O.B	<input type="text" value="04/01/1989 05:15 PM"/>	
SSN	<input type="text" value="510-45-1234"/>	
Phone	<input type="text" value="510-857-466"/>	
Credit card number	<input type="text" value="1234-4567"/>	

Please match the requested format.  
XXX-XXX-XXXX

Pattern(XXXX-XXXX-XXXX-XXXX) is created for the Credit card number. If the pattern is not followed, it would display a message as below.

## Registration Page

First Name:

Goudamy

Last Name:

Sockalingam

email:

goudamy.s@gmail.com

password:

\*\*\*\*\*

Strong

confirm password:

\*\*\*\*\*

D.O.B

04/01/1989

D.O.B with time

12 PM

Local D.O.B

04/01/1989 05:15 PM

SSN

510-45-1234

Phone

510-857-4667

Credit card number

1234-4567-1234-45678

!

Please match the requested format.  
XXXX-XXXX-XXXX-XXXX

### Part 3: Store all the data that has been created so far in Session storage.

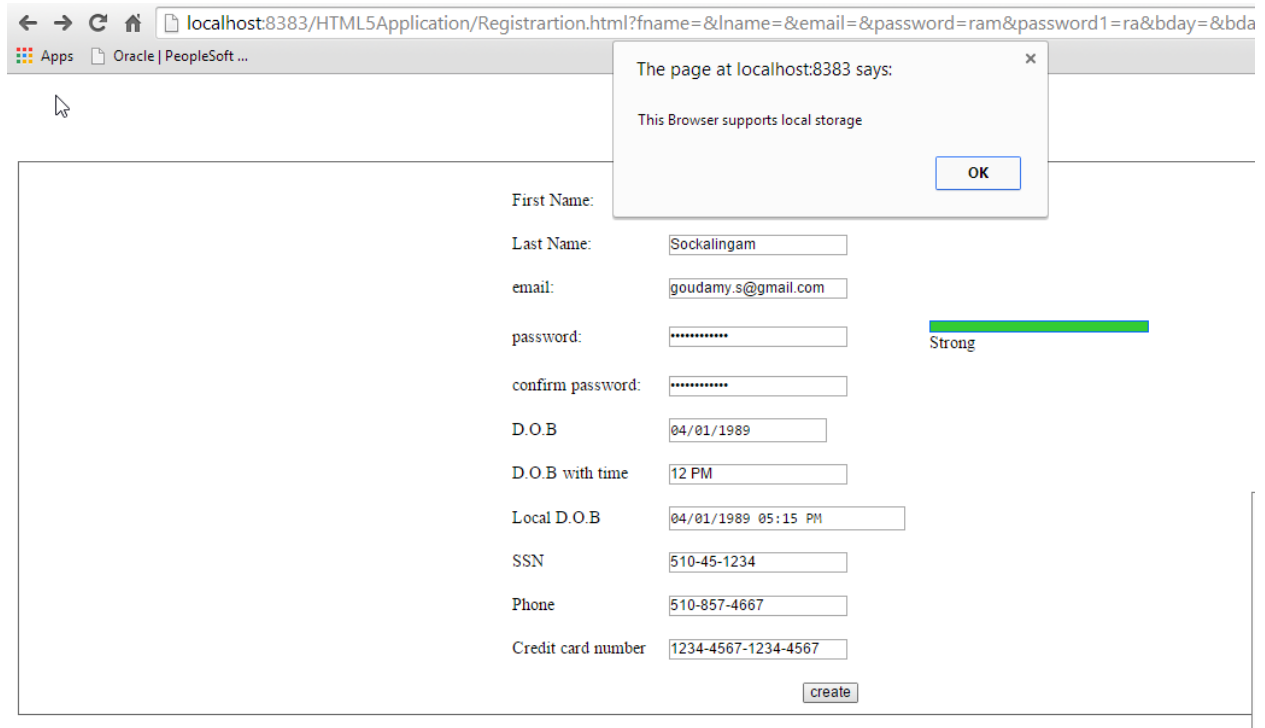
Once the create button is pressed, the values would be saved as session variables. The screenshot below displays the session details that are stored in java script console.

Key	Value
bday	1989-04-01
bday1	12 PM
bday2	1989-04-01T17:15
credit	1234-4567-1234-4567
email	goudamy.s@gmail.com
fname	Goudamy
lname	Sockalingam
passwd	RAMan@#\$\$^&*
phone	510-857-4667
ssn	510-45-1234

#### Part 4: Using “modernizr”, check for local storage across various browsers:

Modernizr is used to check if the browser supports local storage. Modernizr.localstorage is the function that is used for this process. The code is checked for various browsers such as Google chrome, Mozilla Firefox and Internet explorer and all the 3 browsers supports local storage.

#### Chrome:



The screenshot shows a web browser window with the address bar displaying `localhost:8383/HTML5Application/Registrartion.html?fname=&lname=&email=&password=ram&password1=ra&bday=&bda`. A modal dialog box is open, displaying the message: "The page at localhost:8383 says: This Browser supports local storage" with an "OK" button. Below the dialog, a registration form is visible with the following fields and values:

Field	Value
First Name:	
Last Name:	Sockalingam
email:	goudamy.s@gmail.com
password:	*****
confirm password:	*****
D.O.B	04/01/1989
D.O.B with time	12 PM
Local D.O.B	04/01/1989 05:15 PM
SSN	510-45-1234
Phone	510-857-4667
Credit card number	1234-4567-1234-4567

A "create" button is located at the bottom right of the form. A green progress bar next to the password field indicates a "Strong" password.

## Mozilla Firefox:

### Registration Page

First Name:	<input type="text" value="goudamy"/>
Last Name:	<input type="text" value="sockalingam"/>
email:	<input type="text" value=""/>
password:	<input type="password" value=""/>
confirm password:	<input type="password" value=""/>
D.O.B	<input type="text" value="01/04/1989"/>
D.O.B with time	<input type="text" value="7PM"/>
Local D.O.B	<input type="text" value="01/04/1989 7PM"/>
SSN	<input type="text" value="510-85-7765"/>
Phone	<input type="text" value="510-857-4665"/>
Credit card number	<input type="text" value="1234-5678-8901-2345"/>

This Browser supports local storage

OK


## Internet Explorer:

### Registration Page

First Name:	<input type="text" value="Goudamy"/>
Last Name:	<input type="text" value="sockalingam"/>
email:	<input type="text" value="goudamy.s@gmail.com"/>
password:	<input type="password" value=""/>
confirm password:	<input type="password" value=""/>
D.O.B	<input type="text" value="01/04/1989"/>
D.O.B with	<input type="text" value=""/>
Local D.O.B	<input type="text" value=""/>
SSN	<input type="text" value=""/>
Phone	<input type="text" value=""/>
Credit card	<input type="text" value=""/>

create

Message from webpage

 This Browser supports local storage

OK

## Part 5

### Events triggered when

- internet connection is lost
- internet connection is retrieved

The navigator.onLine is the event that is used to check if the internet connection is lost or retrieved.

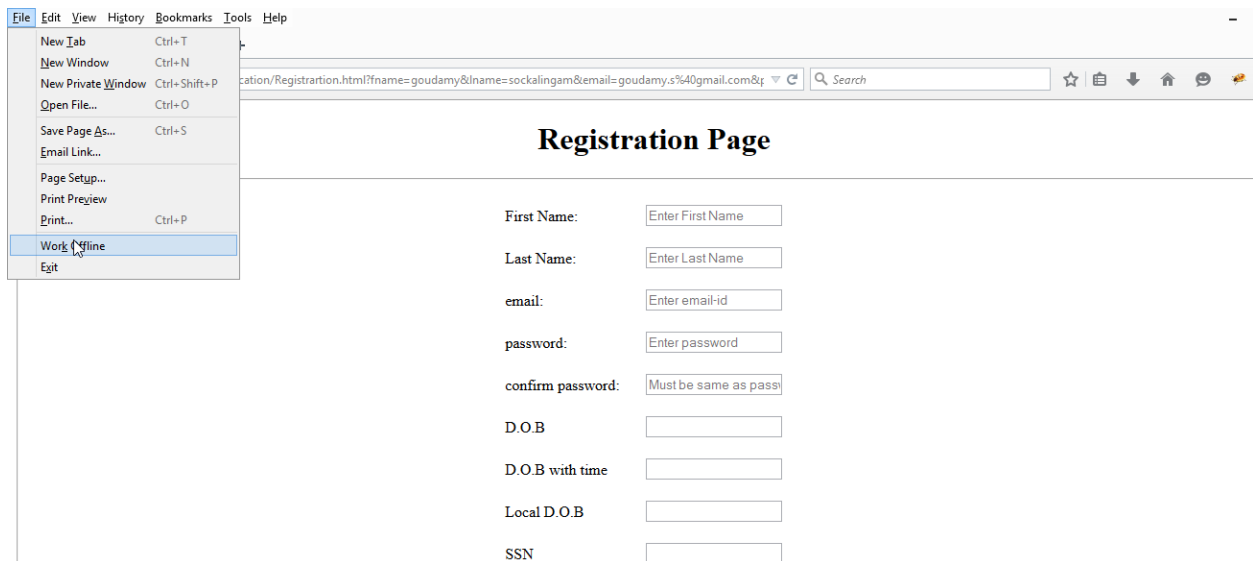
If internet connection is lost then navigator.onLine is false.

If internet connection is retrieved, navigator.onLine becomes true.

## Part 6: If Browser is Offline mode detected, store all the values to local storage.

Browser can be turned to offline mode using File->work offline in Mozilla Firefox.

Navigator.onLine is always displays true in google chrome despite of offline/online mode. So, I tested this feature in Mozilla Firefox.



The screenshot shows the Mozilla Firefox browser interface. The 'File' menu is open, and the 'Work Offline' option is highlighted. The browser's address bar shows a URL for a registration page. The registration form is titled 'Registration Page' and contains several input fields for user information.

Registration Page	
First Name:	<input type="text" value="Enter First Name"/>
Last Name:	<input type="text" value="Enter Last Name"/>
email:	<input type="text" value="Enter email-id"/>
password:	<input type="text" value="Enter password"/>
confirm password:	<input type="text" value="Must be same as pass"/>
D.O.B	<input type="text"/>
D.O.B with time	<input type="text"/>
Local D.O.B	<input type="text"/>
SSN	<input type="text"/>



## Registration Page

First Name:	<input type="text" value="goudamy"/>
Last Name:	<input type="text" value="sockalingam"/>
email:	<input type="text" value="oudamy.s@gmail.com"/>
password:	<input type="password" value="*****"/>
confirm password:	<input type="password" value="*****"/>
D.O.B	<input type="text" value="01/04/1989"/>
D.O.B with time	<input type="text" value="7PM"/>
Local D.O.B	<input type="text" value="01/04/1989 7PM"/>
SSN	<input type="text" value="510-85-7765"/>
Phone	<input type="text" value="510-857-4665"/>
Credit card number	<input type="text" value="1234-5678-8901-2345"/>
<input type="button" value="create"/>	

When the browser is turned to offline mode, it stores the values in local storage.

The screenshot shows the Chrome DevTools DOM panel with the 'localStorage' property highlighted. The 'localStorage' object contains 9 items, including 'bday1', 'phone', 'email', and 'password1'. The 'password1' value is a long, encoded string.

Property	Value
document	Document Registration.html?fname=goudamy&lname=sockalingam&email=goudamy.s@gmail.com&password=RAMan%40%23%24%25%5E%26&password1=RAMan%40%23%24%25%5E%26&bday=01%2F04%2F1989&bday1=7PM&bday2=01%2F04%2F1989+7PM&ssn=510-85-7765&phone=510-857-4665&credit=1234-5678-8901-2345&submit=create
external	External ( AddSearchProvider=AddSearchProvider(), IsSearchProviderInstalled=IsSearchProviderInstalled(), addSearchEngine=addSearchEngine() )
frameElement	null
frames	Window Registration.html?fname=.8901-2345&submit=create
fullScreen	false
gform	form
history	3 history entries
indexedDB	IDBFactory { open=open(), deleteDatabase=deleteDatabase(), cmp=cmp() }
innerHeight	161
innerWidth	1366
length	0
localStorage	9 items in Storage bday1="7 PM", phone="510-857-4665", email="goudamy@gmail.com", more...
location	http://localhost:8383/HTML5Application/Registration.html?fname=goudamy&lname=sockalingam&email=goudamy.s@gmail.com&password=RAMan%40%23%24%25%5E%26&password1=RAMan%40%23%24%25%5E%26&bday=01%2F04%2F1989&bday1=7PM&bday2=01%2F04%2F1989+7PM&ssn=510-85-7765&phone=510-857-4665&credit=1234-5678-8901-2345&submit=create ( href="http://localhost:8383/HT...8901-2345&submit=create", origin="http://localhost:8383", protocol="http:", more... )
locationbar	BarProp { visible=true }
menubar	BarProp { visible=true }
mozAnimationStartTime	1424979965051
mozIndexedDB	IDBFactory { open=open(), deleteDatabase=deleteDatabase(), cmp=cmp() }
mozInnerScreenX	0

The screenshot shows the Chrome DevTools localStorage panel. The 'localStorage' object contains 9 items, including 'bday', 'bday1', 'bday2', 'email', 'fname', 'lname', 'password1', 'phone', and 'ssn'. The 'password1' value is a long, encoded string.

Property	Value
bday	"01-04-1989"
bday1	"7 PM"
bday2	"01-04-1989 7PM"
email	"goudamy@gmail.com"
fname	"goudamy"
lname	"sockalingam"
password1	"RAMan%40%23%24%25%5E%26"
phone	"510-857-4665"
ssn	"1234-5678-8901-2345"
__proto__	StoragePrototype { key=key(), getItem=getItem(), setItem=setItem(), more... }