

SPRINT 3

Deploying the app in local host :

```
dbapp.py x
1 import ibm_db
2
3 hostname="fbd88901-ebdb-4a4f-a32e-9822b9fb237b.clogj3sd0tgtu0lqde00.databases.appdomain.cloud"
4 uid="gxr37142"
5 pwd="JRrMYR3jeR3ykbXx"
6 driver="[IBM DB2 ODBC DRIVER]"
7 db="bludb"
8 port="32731"
9 protocol="TCP/IP"
10 cert="Certificate.crt"
11 dsn=(
12     "DATABASE={0};"
13     "HOSTNAME={1};"
14     "PORT={2};"
15     "UID={3};"
16     "SECURITY=SSL;"
17     "SSL ServerCertificate={4};"
18     "PWD={5};"
19 ).format(db, hostname, port, uid, cert, pwd)
20 print(dsn)
21 try:
22     db2=ibm_db.connect(dsn, "", "")
23     print("Connected to data base")
24 except:
25     print("Unable to connect", ibm_db.conn_errormsg())
```

```
C:\Windows\System32\cmd.e x + v
Microsoft Windows [Version 10.0.22621.819]
(c) Microsoft Corporation. All rights reserved.

C:\Users\MADHUBALAN\OneDrive\Desktop\data base>python dbapp.py
DATABASE=bludb;HOSTNAME= 54a2f15b-5c0f-46df-8954-7e38e612c2bd.clogj3sd0tgtu0lqde00.databases.appdomain.cloud;PORT=32733;
UID=tct88303 ;SECURITY=SSL;SSL ServerCertificate=Certificate.crt;PWD=C5yDGA2a2q6Lng6U;
Connected to data base

C:\Users\MADHUBALAN\OneDrive\Desktop\data base>|
```

REGISTRATION PAGE:



The image shows a 'Register' form overlay on a background of a person's hand typing on a laptop. The form is titled 'Register' and contains the following fields and elements:

- Email input field
- Mobile number input field
- Password input field
- Confirm Password input field
- I agree to the terms checkbox
- Register button (blue)
- Already have an account ? Login link

IBM Db2 on Cloud

Load Data Load History **Tables** Views Indexes Aliases MQTs Sequences Application objects

GXR37142.USER Back

Export to CSV

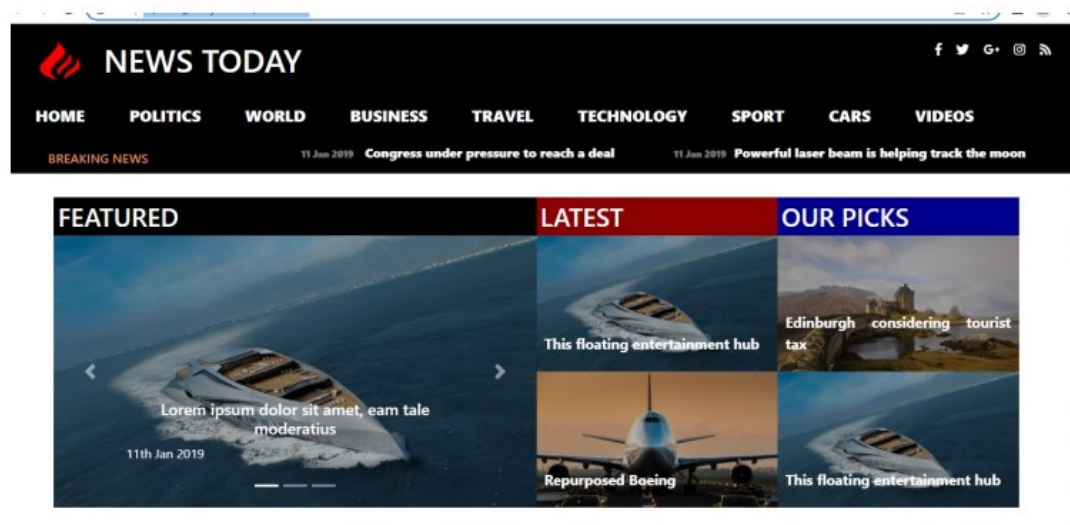
USERNAME	EMAIL	PASSWORD
chandra kumar	ckchandru6302@gmail.com	2002
hariprasad	haridhoni0077@gmail.com	45123
jai praveen	jamunasomya@gmail.com	23451
madhu balan	madhubalan5star@gmail.com	madhu@123
mohammed khubabe	mohammedkhubabe777@gmail.com	54321
nikil	knikal156@gmail.com	77777
rahmani	rahmanis15702@email.com	12345

Activate Windows
Go to Settings to activate Windows.

LOGIN PAGE:



HOME PAGE:



Code :

```
from
cloudant.clie
nt import
Cloudant

from cloudant.error import CloudantException
from cloudant.result import Result, ResultByKey
from flask import Flask, render_template, request
from newsapi import NewsApiClient

serviceUsername = "50ab9a31-993d-4eaf-abf1-21c6f737b08c-blue
servicePassword =
"6864bc1c41f5701310846a0a80700140c138e7c82b8c7f678ce5ab877f8b90b9"
serviceURL = "https://50ab9a31-993d-4eaf-abf1-21c6f737b08c-
blue
mix:6864bc1c41f5701310846a0a80700140c138e7c82b8c7f678ce5ab877f8b90b9@50ab
9a31-993d-4eaf-abf1-21c6f737b08c-blue
mix.cloudantnosql
db.appdomain.cloud"

client = Cloudant(serviceUsername, servicePassword, url=serviceURL)
client.connect()

app = Flask(__name__)

@app.route("/news")
def news():
    api_key = '4f625485702d4929a546bcf4eb9d5c79'

    newsapi = NewsApiClient(api_key=api_key)

    top_headlines = newsapi.get_top_headlines(sources = "the-verge")
    all_articles = newsapi.get_everything(sources = "the-verge")

    t_articles = top_headlines['articles']
    a_articles = all_articles['articles']

    news = []
    desc = []
    img = []
    p_date = []
    url = []

    for i in range (len(t_articles)):
        main_article = t_articles[i]
```

```

news.append(main_article['title'])
desc.append(main_article['description'])
img.append(main_article['urlToImage'])
p_date.append(main_article['publishedAt'])
url.append(main_article['url'])

contents = zip( news,desc,img,p_date,url)

news_all = []
desc_all = []
img_all = []
p_date_all = []
url_all = []

for j in range(len(a_articles)):
    main_all_articles = a_articles[j]

    news_all.append(main_all_articles['title'])
    desc_all.append(main_all_articles['description'])
    img_all.append(main_all_articles['urlToImage'])
    p_date_all.append(main_all_articles['publishedAt'])
    url_all.append(main_article['url'])

all = zip( news_all,desc_all,img_all,p_date_all,url_all)

return render_template('home.html',all = all)

@app.route("/search",methods = ['POST', 'GET'])
def searchFunct():
    inputText = request.form['nm']
    api_key = '4f625485702d4929a546bcf4eb9d5c79'

    newsapi = NewsApiClient(api_key=api_key)

    top_headlines = newsapi.get_top_headlines(sources="bbc-news")
    all_articles = newsapi.get_everything(q=inputText)

    t_articles = top_headlines['articles']
    a_articles = all_articles['articles']

    news = []
    desc = []
    img = []
    p_date = []
    url = []

```

```

for i in range (len(t_articles)):
    main_article = t_articles[i]

    news.append(main_article['title'])
    desc.append(main_article['description'])
    img.append(main_article['urlToImage'])
    p_date.append(main_article['publishedAt'])
    url.append(main_article['url'])

    contents = zip( news,desc,img,p_date,url)

    news_all = []
    desc_all = []
    img_all = []
    p_date_all = []
    url_all = []

    for j in range(len(a_articles)):
        main_all_articles = a_articles[j]

        news_all.append(main_all_articles['title'])
        desc_all.append(main_all_articles['description'])
        img_all.append(main_all_articles['urlToImage'])
        p_date_all.append(main_all_articles['publishedAt'])
        url_all.append(main_article['url'])

    all = zip( news_all,desc_all,img_all,p_date_all,url_all)

    return render_template('home.html', all = all)

def addNewUser(userName,userEmail,userPassword):
    jsondata = {}
    jsondata["userName"] = str(userName)
    jsondata["userEmail"] = str(userEmail)
    jsondata["userPassword"] = str(userPassword)

    myDataBase = client['database1']
    newDocument = myDataBase.create_document(jsondata)

def authenticate(userName,userEmail):
    myDataBase = client['database1']
    result_collection = Result(myDataBase.all_docs, include_docs=True)
    for data in result_collection:

        if data['doc']['userName'] == str(userName):

```

```

return True
if data['doc']['userEmail'] == str(userEmail):
return True
return False

def authenticateLogin(userEmail,userPassword):
myDataBase = client['database1']
result_collection = Result(myDataBase.all_docs, include_docs=True)

for data in result_collection:
if data['doc']['userPassword'] == str(userPassword) and
data['doc']['userEmail'] == str(userEmail):
return True
return False

@app.route("/login",methods = ['POST', 'GET'])
def loginUser():

userEmail = request.form.get("email")
userPassword = request.form.get("pswd")

if(authenticateLogin(userEmail,userPassword)):
return news()
return render_template("index.html")

@app.route("/register",methods = ['POST', 'GET'])
def registerUserData():
userName = request.form.get("un")
userEmail = request.form.get("ue")
userPassword = request.form.get("up")

print(userEmail,userName,userPassword)
if(authenticate(userName=userName,userEmail=userEmail)):
return render_template("index.html")
addNewUser(userName,userEmail,userPassword)
return news()

@app.route("/")
def home():
return render_template("index.html")

@app.route("/contact")
def hh():
return render_template("contact.html")

```

```
if __name__ == '__main__':  
    app.run(debug=True)
```