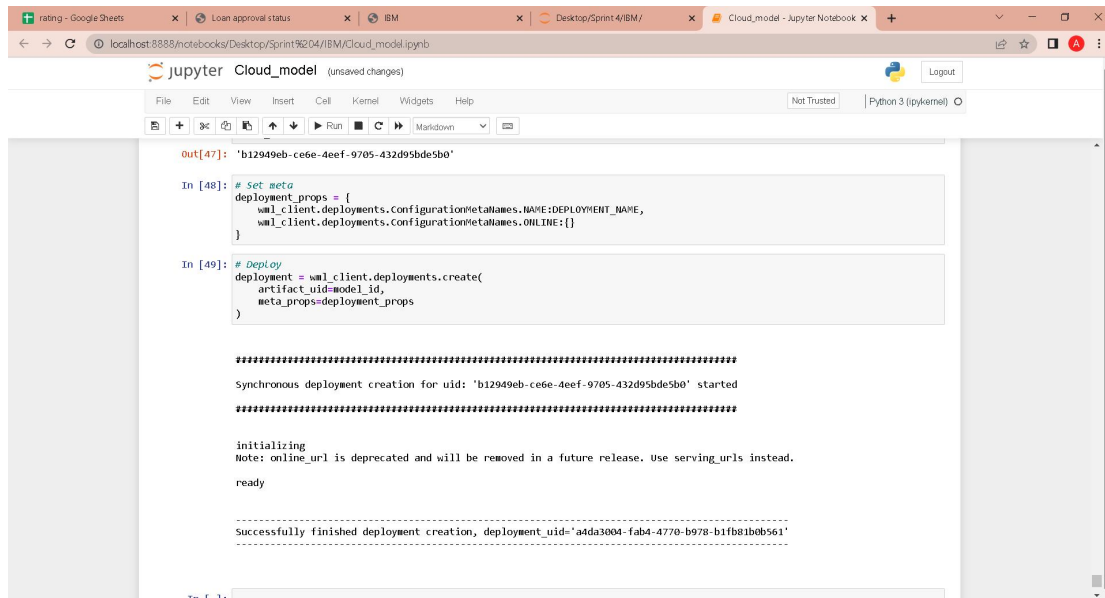


TESTED SCREENSHOTS OF SPRINT 4

IBM cloud deployment:



```
out[47]: 'b12949eb-ce6e-4eef-9705-432d95bde5b0'

In [48]: # Set meta
deployment_props = {
    wml_client.deployments.ConfigurationMetaNames.NAME: DEPLOYMENT_NAME,
    wml_client.deployments.ConfigurationMetaNames.ONLINE: {}
}

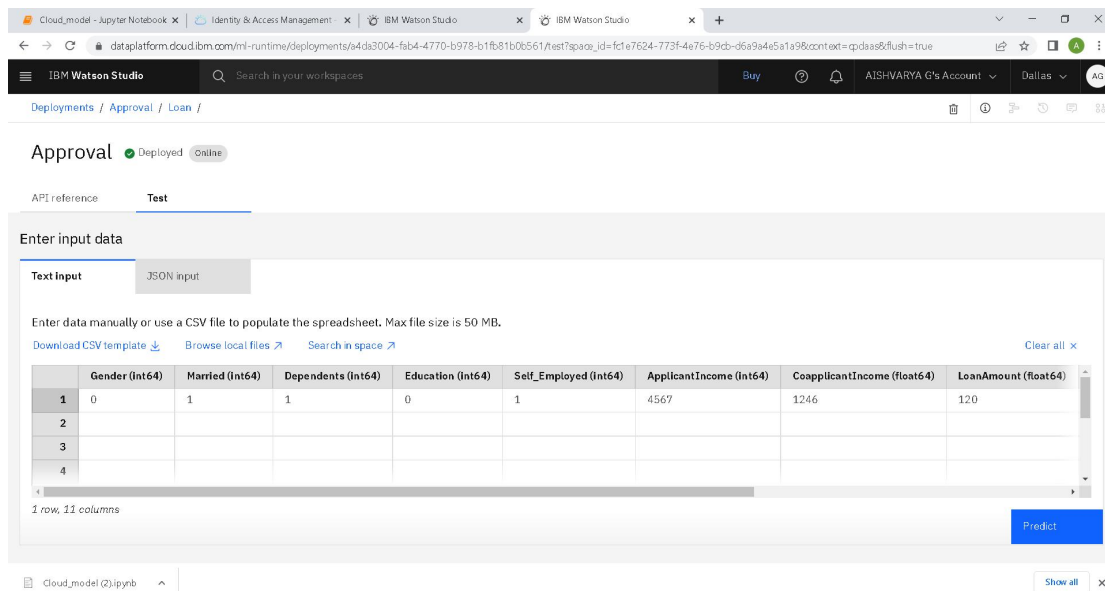
In [49]: # Deploy
deployment = wml_client.deployments.create(
    artifact_uid=model_id,
    meta_props=deployment_props
)

#####
Synchronous deployment creation for uid: 'b12949eb-ce6e-4eef-9705-432d95bde5b0' started
#####

initializing
Note: online_url is deprecated and will be removed in a future release. Use serving_urls instead.
ready

-----
Successfully finished deployment creation, deployment_uid='adda3004-fab4-4770-b978-b1fb81b0b561'
```

Testing:



Approval Deployed Online

API reference **Test**

Enter input data

Text input JSON input

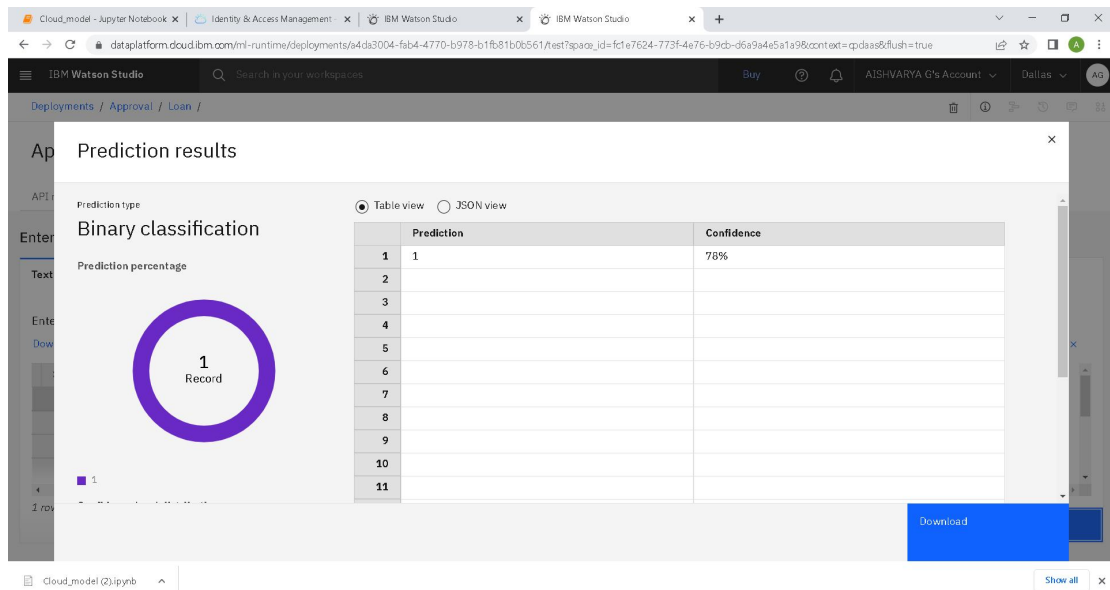
Enter data manually or use a CSV file to populate the spreadsheet. Max file size is 50 MB.

[Download CSV template](#) [Browse local files](#) [Search in space](#) [Clear all](#)

	Gender (int64)	Married (int64)	Dependents (int64)	Education (int64)	Self_Employed (int64)	ApplicantIncome (int64)	CoapplicantIncome (float64)	LoanAmount (float64)
1	0	1	1	0	1	4567	1246	120
2								
3								
4								

1 row, 11 columns

[Predict](#)



Running the application through deployed model in IBM cloud:

```
1 from flask import render_template, Flask, request
2 import numpy as np
3 import pickle
4 import requests
5
6 # NOTE: you must manually set API_KEY below using information retrieved from your IBM Cloud account.
7 API_KEY = "pyswabyz3tufvrxm2hld7dsoinqu4i1sv0mk8y"
8 token_response = requests.post("https://iam.cloud.ibm.com/identity/token", data={"apikey": API_KEY, "grant_type": "urn:ibm:params:oauth:grant-type:apikey"})
9 ml_token = token_response.json()["access_token"]
10
11 header = {'content-type': 'application/json', 'Authorization': 'Bearer ' + ml_token}
12 app = Flask(__name__, template_folder='templates')
13
14 @app.route('/')
15 def home():
16     return render_template('home.html')
17 @app.route('/login.html')
18 def login():
19     return render_template('login.html')
20 @app.route('/procedure.html')
21 def procedure():
22     return render_template('procedure.html')
23 @app.route('/bank_login.html')
24 def bank():
25     return render_template('bank_login.html')
```

WARNING: This is a development server. Do not use it in a production deployment. Use a production WSGI server instead.

* Running on http://127.0.0.1:5000

Press CTRL+C to quit

* Restarting with stat

C:\Users\Aishwarya\AppData\Local\Programs\Python\Python311\Lib\site-packages\sklearn\base.py:329: UserWarning: Trying to unpickle estimator DecisionTreeClassifier from version 1.1.2 when using version 1.1.3. This might lead to breaking code or invalid results. Use at your own risk. For more info please refer to: https://scikit-learn.org/stable/model_persistence.html#security-maintainability-limitations

warnings.warn(C:\Users\Aishwarya\AppData\Local\Programs\Python\Python311\Lib\site-packages\sklearn\base.py:329: UserWarning: Trying to unpickle estimator RandomForestClassifier from version 1.1.2 when using version 1.1.3. This might lead to breaking code or invalid results. Use at your own risk. For

Output:

LOAN APPROVAL STATUS

Sorry You are not eligible for loan



Please provide your feedback

