```
import pandas as pd
import numpy as np
from sklearn.linear_model import LogisticRegression

dia = pd.read_excel("/content/drive/MyDrive/diabetes (1).xlsx")
dia.head()
```

```
{"summary":"{\n  \"name\": \"dia\",\n  \"rows\": 768,\n  \"fields\":
[\n    {\n      \"column\": \"preg\",\n      \"properties\": {\n
\"dtype\": \"number\",\n        \"std\": 3,\n        \"min\": 0,\n
\"max\": 17,\n        \"num_unique_values\": 17,\n        \"samples\":
[\n          6,\n          1,\n          3\n        ],\n
\"semantic_type\": \"\",\n        \"description\": \"\"\n      }\
n    },\n    {\n      \"column\": \"plas\",\n      \"properties\": {\n
\"dtype\": \"number\",\n        \"std\": 31,\n        \"min\": 0,\n
\"max\": 199,\n        \"num_unique_values\": 136,\n
\"samples\": [\n          151,\n          101,\n          112\n
],\n        \"semantic_type\": \"\",\n        \"description\": \"\"\n
}\n    },\n    {\n      \"column\": \"pres\",\n      \"properties\":
{\n        \"dtype\": \"number\",\n        \"std\": 19,\n
\"min\": 0,\n        \"max\": 122,\n        \"num_unique_values\":
47,\n        \"samples\": [\n          86,\n          46,\n
85\n        ],\n        \"semantic_type\": \"\",\n
\"description\": \"\"\n      }\n    },\n    {\n      \"column\":
\"skin\",\n      \"properties\": {\n        \"dtype\": \"number\",\n
\"std\": 15,\n        \"min\": 0,\n        \"max\": 99,\n
\"num_unique_values\": 51,\n        \"samples\": [\n          7,\n
12,\n          48\n        ],\n        \"semantic_type\": \"\",\n
\"description\": \"\"\n      }\n    },\n    {\n      \"column\":
\"insu\",\n      \"properties\": {\n        \"dtype\": \"number\",\n
\"std\": 115,\n        \"min\": 0,\n        \"max\": 846,\n
\"num_unique_values\": 186,\n        \"samples\": [\n          52,\n
41,\n          183\n        ],\n        \"semantic_type\": \"\",\n
\"description\": \"\"\n      }\n    },\n    {\n      \"column\":
\"mass\",\n      \"properties\": {\n        \"dtype\": \"number\",\n
\"std\": 7.8841603203754405,\n        \"min\": 0.0,\n        \"max\":
67.1,\n        \"num_unique_values\": 248,\n        \"samples\": [\n
19.9,\n          31.0,\n          38.1\n        ],\n
\"semantic_type\": \"\",\n        \"description\": \"\"\n      }\
n    },\n    {\n      \"column\": \"pedi\",\n      \"properties\": {\n
\"dtype\": \"number\",\n        \"std\": 0.33132859501277484,\n
\"min\": 0.078,\n        \"max\": 2.42,\n
\"num_unique_values\": 517,\n        \"samples\": [\n          1.731,\
n          0.426,\n          0.138\n        ],\n
\"semantic_type\": \"\",\n        \"description\": \"\"\n      }\
n    },\n    {\n      \"column\": \"age\",\n      \"properties\": {\n
\"dtype\": \"number\",\n        \"std\": 11,\n        \"min\": 21,\n
\"max\": 81,\n        \"num_unique_values\": 52,\n        \"samples\":
[\n          60,\n          47,\n          72\n        ],\n
\"semantic_type\": \"\",\n        \"description\": \"\"\n      }\
n    },\n    {\n      \"column\": \"class\",\n      \"properties\": {\
n        \"dtype\": \"category\",\n        \"num_unique_values\": 2,\n
\"samples\": [\n          \"tested_negative\",\n
\"tested_positive\"\n        ],\n        \"semantic_type\": \"\",\n
\"description\": \"\"\n      }\n    }\n  ]\
n}","type":"dataframe","variable_name":"dia"}
```

```
dia.isnull().sum()
```

```
preg     0
plas     0
pres     0
skin     0
insu     0
mass     0
pedi     0
age      0
class    0
dtype: int64
```

```
ind = dia[['age','mass','insu','plas']]
dep = dia['class']
```

```
Logr = LogisticRegression()
```

```
Logr.fit(ind,dep)
```

```
LogisticRegression()
```

```
age = int(input("Enter the age:"))
mass = int(input("Enter the mass:"))
insulin = int(input("Enter the insulin level:"))
plasma = int(input("Enter the plasma level:"))
pred = Logr.predict([[age,mass,insulin,plasma]])
print(pred)
```

```
Enter the age:23
Enter the mass:78
Enter the insulin level:99
Enter the plasma level:32
['tested_negative']
```

```
/usr/local/lib/python3.12/dist-packages/sklearn/utils/
validation.py:2739: UserWarning: X does not have valid feature names,
but LogisticRegression was fitted with feature names
  warnings.warn(
```

```
Logr.score(ind,dep)
```

```
0.7669270833333334
```