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

```
train_data = pd.read_csv('/content/wheet_train.csv')
tr = train_data.drop(['ID'],axis=1)
tr.head()
```

	area	perimeter	compactness	kernelLength	kernelWidth	asymmetryCoefficient	ke
0	18.59	16.05	0.9066	6.037	3.860	6.001	
1	11.18	12.72	0.8680	5.009	2.810	4.051	
2	15.99	14.89	0.9064	5.363	3.582	3.336	
3	15.38	14.90	0.8706	5.884	3.268	4.462	
4	19.15	16.45	0.8890	6.245	3.815	3.084	

```
y = tr['Type']
X = tr.drop(['Type'], axis=1)
```

```
test_data = pd.read_csv('/content/wheet_test.csv')
test_data.head()
```

	ID	area	perimeter	compactness	kernelLength	kernelWidth	asymmetryCoefficient
0	1	18.85	16.17	0.9056	6.152	3.806	2.843
1	2	11.34	12.87	0.8596	5.053	2.849	3.347
2	3	14.86	14.67	0.8676	5.678	3.258	2.129
3	4	12.67	13.32	0.8977	4.984	3.135	2.300
4	5	11.82	13.40	0.8274	5.314	2.777	4.471

```
tst = test_data.drop(['ID'],axis=1)
```

```
log_reg = LogisticRegression()
log_reg = log_reg.fit(X,y)
```

```
/usr/local/lib/python3.7/dist-packages/sklearn/linear_model/_logistic.py:818: Converg
STOP: TOTAL NO. of ITERATIONS REACHED LIMIT.

Increase the number of iterations (max_iter) or scale the data as shown in:
    https://scikit-learn.org/stable/modules/preprocessing.html
Please also refer to the documentation for alternative solver options:
    https://scikit-learn.org/stable/modules/linear_model.html#logistic-regression
  extra_warning_msg=_LOGISTIC_SOLVER_CONVERGENCE_MSG,
```

```
y_test = log_reg.predict(tst)
print(y_test)
```

```
[2 3 1 1 3 2 1 2 1 2 3 1 2 2 3 3 2 2 3 1 3 1 3 2 3 1 3 1 2 2 1 2 2 1 3 2 2
 2 2 2 3 3 3 3 1 2 3 1 3 2 3 1 1 3 3 1 3 2 2 1 1 2 2 2 1 1 2 3 1 1]
```

```
id = test_data['ID']
```

```
import csv
submission = open("log_regr.csv","w")
sub_file = csv.writer(submission)
sub_file.writerow(['ID','Type'])
for i in range(0,len(y_test)):
    sub_file.writerow([str(id[i]),str(y_test[i])])

submission.close()
```