Slip 1 - Q1: Apriori Algorithm

```
from mlxtend.frequent_patterns import apriori, association_rules
import pandas as pd
df = pd.read_csv("groceries.csv")
basket = df.stack().str.get_dummies().sum(level=0)
freq = apriori(basket, min_support=0.25, use_colnames=True)
rules = association_rules(freq, metric="lift", min_threshold=1)
print(freq)
print(rules)
```

Slip 1 - Q2: Iris Scatter Plot with Encoding

```
import seaborn as sns
import pandas as pd
from sklearn.datasets import load_iris
import matplotlib.pyplot as plt
iris = load_iris()
df = pd.DataFrame(iris.data, columns=iris.feature_names)
df['species'] = iris.target
sns.scatterplot(x=df.iloc[:,0], y=df.iloc[:,1], hue=df['species'])
plt.show()
```

Slip 2 - Q1: Simple Linear Regression (House Price)

```
import pandas as pd
from sklearn.linear_model import LinearRegression
df = pd.read_csv("house.csv").dropna()
model = LinearRegression().fit(df[['area']], df['price'])
print(model.predict([[2000]]))
```

Slip 2 - Q2: Agglomerative Clustering

```
import pandas as pd
from sklearn.cluster import AgglomerativeClustering
df = pd.read_csv("wholesale.csv")
labels = AgglomerativeClustering(n_clusters=3).fit_predict(df)
print(labels)
```

Slip 3 - Q1: Multiple Linear Regression

```
import pandas as pd
from sklearn.model_selection import train_test_split
from sklearn.linear_model import LinearRegression
df = pd.read_csv("house.csv")
X = df[['area','rooms']]
y = df['price']
Xtr, Xte, ytr, yte = train_test_split(X, y, test_size=0.2)
model = LinearRegression().fit(Xtr, ytr)
print(model.score(Xte, yte))
```

Slip 3 - Q2: Logistic Regression

```
import pandas as pd
from sklearn.linear_model import LogisticRegression
df = pd.read_csv("crash.csv")
model = LogisticRegression().fit(df[['age','speed']], df['survived'])
print(model.predict([[30,60]]))
```

Slip 4 - Q1: K-Means Clustering (Mall Customers)

```
import pandas as pd
from sklearn.cluster import KMeans
df = pd.read_csv("mall_customers.csv")
km = KMeans(n_clusters=5).fit(df[['Annual Income (k$)','Spending Score (1-100)']])
print(km.labels_)
```

Slip 4 - Q2: Simple Linear Regression (House Price)

```
import pandas as pd
from sklearn.linear_model import LinearRegression
df = pd.read_csv("house.csv")
model = LinearRegression().fit(df[['area']], df['price'])
print(model.predict([[1500]]))
```

Slip 5 - Q1: Multiple Linear Regression (Fuel Consumption)

```
import pandas as pd
from sklearn.linear_model import LinearRegression
df = pd.read_csv("fuel.csv")
model = LinearRegression().fit(df[['Engine Size','Cylinders']], df['Fuel Consumption'])
print(model.coef_)
```

Slip 5 - Q2: kNN on Iris Dataset

```
from sklearn.datasets import load_iris
from sklearn.neighbors import KNeighborsClassifier
iris = load_iris()
knn = KNeighborsClassifier(n_neighbors=3).fit(iris.data, iris.target)
print(knn.predict([[5,3,1,0]]))
```

Slip 6 - Q1: Polynomial Regression (Boston Housing)

```
import pandas as pd
from sklearn.preprocessing import PolynomialFeatures
from sklearn.linear_model import LinearRegression
df = pd.read_csv("boston.csv")
poly = PolynomialFeatures(2)
Xp = poly.fit_transform(df[['RM']])
model = LinearRegression().fit(Xp, df['MEDV'])
print(model.predict(poly.transform([[6]])))
```

Slip 6 - Q2: K-means for Employee Income

```
import pandas as pd
from sklearn.cluster import KMeans
df = pd.read_csv("employees.csv").dropna()
print(KMeans(3).fit_predict(df[['income']]))
```

Slip 7 - Q1: Salary Prediction

```
import pandas as pd
from sklearn.linear_model import LinearRegression
df = pd.read_csv("salary_positions.csv")
model = LinearRegression().fit(df[['Level']], df['Salary'])
print(model.predict([[11],[12]]))
```

Slip 7 - Q2: Naive Bayes Weather Dataset

```
import pandas as pd
from sklearn.naive_bayes import GaussianNB
df = pd.read_csv("weather.csv")
model = GaussianNB().fit(df[['Humidity','Temperature']], df['Play'])
print(model.predict([[70,25]]))
```

Slip 8 - Q1: 20 Newsgroups Classification

```
from sklearn.datasets import fetch_20newsgroups
from sklearn.feature_extraction.text import CountVectorizer
from sklearn.naive_bayes import MultinomialNB
data = fetch_20newsgroups(subset='train')
vec = CountVectorizer().fit(data.data)
X = vec.transform(data.data)
model = MultinomialNB().fit(X, data.target)
print("Model Trained")
```

Slip 8 - Q2: Decision Tree (Play Tennis)

```
import pandas as pd
from sklearn.tree import DecisionTreeClassifier
df = pd.read_csv("tennis.csv")
X = pd.get_dummies(df.drop('play',axis=1))
y = df['play']
model = DecisionTreeClassifier().fit(X,y)
print(model.predict(X.iloc[:1]))
```

Slip 9 - Q1: Ridge & Lasso Regression

```
import pandas as pd
from sklearn.linear_model import Ridge, Lasso
df = pd.read_csv("boston_houses.csv")
ridge = Ridge().fit(df[['RM']], df['Price'])
lasso = Lasso().fit(df[['RM']], df['Price'])
print(ridge.predict([[5]]))
print(lasso.predict([[5]]))
```

Slip 9 - Q2: Linear SVM

```
import pandas as pd
from sklearn.svm import SVC
df = pd.read_csv("UniversalBank.csv")
model = SVC(kernel='linear').fit(df[['Age','Income']], df['Personal Loan'])
print(model.predict([[30,50000]]))
```

Slip 10 - Q1: PCA on Iris

```
from sklearn.datasets import load_iris
from sklearn.decomposition import PCA
iris = load_iris()
pca = PCA(2)
print(pca.fit_transform(iris.data)[:5])
```

Slip 10 - Q2: Scatter Plot + Encoding

```
import seaborn as sns
import pandas as pd
from sklearn.datasets import load_iris
iris = load_iris()
df = pd.DataFrame(iris.data, columns=iris.feature_names)
df['species'] = iris.target
sns.scatterplot(x=df.iloc[:,0], y=df.iloc[:,1], hue=df['species'])
```