In [ ]:

```
#https://medium.com/towards-data-science/pycaret-and-streamlit-how-to-create-and-deploy-data-science-web-app-273
import pandas as pd
import numpy as np
```

In [ ]:

```
shop_df = pd.read_csv('transactions.csv')
```

In [ ]:

```
shop_df.head()
```

Out[ ]:

	Shop	Order_Date_FK	ProductCode	OriginalSaleAmountInclVAT	CustomerID	SaleDocumentNumber	R
0	36	20210730	-1968361059464632550	99.95	-2190786785520839526	23995792	
1	58	20210730	-1745497788861904687	21.95	-3499949798555834600	23995794	
2	49	20210730	-4482110002029860032	69.95	-7841417233849937172	23995797	
3	49	20210730	-9146072753857637799	99.95	-7841417233849937172	23995797	
4	15	20210730	-8829485741699345914	49.95	7033964597104594686	23995798	

In [ ]:

```
from pycaret.classification import *
exp_clf01 = setup(data = shop_df, target = 'Returned', session_id = 123)
```

	Description	Value
0	Session id	123
1	Target	Returned
2	Target type	Binary
3	Original data shape	(1759889, 12)
4	Transformed data shape	(1759889, 12)
5	Transformed train set shape	(1231922, 12)
6	Transformed test set shape	(527967, 12)
7	Numeric features	11
8	Preprocess	True
9	Imputation type	simple
10	Numeric imputation	mean
11	Categorical imputation	mode
12	Fold Generator	StratifiedKFold
13	Fold Number	10
14	CPU Jobs	-1
15	Use GPU	False
16	Log Experiment	False
17	Experiment Name	clf-default-name
18	USI	9471

```
In [ ]: exp_clf102 = setup(data = shop_df, target = 'Returned', session_id=123, normalize = True, transformation = True)
```

	Description	Value
0	Session id	123
1	Target	Returned
2	Target type	Binary
3	Original data shape	(1759889, 12)
4	Transformed data shape	(1759889, 12)
5	Transformed train set shape	(1231922, 12)
6	Transformed test set shape	(527967, 12)
7	Numeric features	11
8	Preprocess	True
9	Imputation type	simple
10	Numeric imputation	mean
11	Categorical imputation	mode
12	Transformation	True
13	Transformation method	yeo-johnson
14	Normalize	True
15	Normalize method	zscore
16	Fold Generator	StratifiedKFold
17	Fold Number	10
18	CPU Jobs	-1

19	Use GPU	False
20	Log Experiment	False
21	Experiment Name	clf-default-name
22	USI	0763

```
In [ ]: best = compare_models()
```

	Model	Accuracy	AUC	Recall	Prec.	F1	Kappa	MCC	TT (Sec)
rf	Random Forest Classifier	0.9685	0.4535	0.3891	0.6543	0.4879	0.4728	0.4899	147.5240
et	Extra Trees Classifier	0.9674	0.4902	0.3927	0.6226	0.4815	0.4656	0.4789	53.6090
lr	Logistic Regression	0.9615	0.6737	0.0000	0.0000	0.0000	0.0000	0.0000	9.5850
nb	Naive Bayes	0.9615	0.5760	0.0000	0.0000	0.0000	0.0000	0.0000	6.7990
svm	SVM - Linear Kernel	0.9615	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	7.8040
ridge	Ridge Classifier	0.9615	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	7.2270
ada	Ada Boost Classifier	0.9615	0.5809	0.0000	0.1000	0.0000	0.0000	0.0013	29.0470
gbc	Gradient Boosting Classifier	0.9615	0.4717	0.0001	0.2500	0.0002	0.0002	0.0043	68.4330
lda	Linear Discriminant Analysis	0.9615	0.5000	0.0000	0.0000	0.0000	0.0000	0.0000	6.3720
dummy	Dummy Classifier	0.9615	0.5000	0.0000	0.0000	0.0000	0.0000	0.0000	6.1080
lightgbm	Light Gradient Boosting Machine	0.9614	0.6161	0.0005	0.2242	0.0010	0.0008	0.0085	9.6970
qda	Quadratic Discriminant Analysis	0.9609	0.5000	0.0058	0.2060	0.0112	0.0092	0.0286	7.7660
knn	K Neighbors Classifier	0.9582	0.5011	0.0673	0.3061	0.1104	0.0978	0.1285	44.4090
dt	Decision Tree Classifier	0.9567	0.5024	0.4372	0.4378	0.4375	0.4150	0.4150	11.6490

In [ ]:

```
rf_model = create_model('rf')
```

	Accuracy	AUC	Recall	Prec.	F1	Kappa	MCC
Fold							
0	0.9684	0.4143	0.3765	0.6558	0.4784	0.4633	0.4824
1	0.9683	0.5083	0.3822	0.6502	0.4814	0.4662	0.4838
2	0.9688	0.4148	0.3983	0.6553	0.4955	0.4803	0.4963
3	0.9690	0.4669	0.3973	0.6621	0.4966	0.4816	0.4984
4	0.9687	0.4640	0.3939	0.6574	0.4926	0.4775	0.4943
5	0.9686	0.4712	0.3726	0.6657	0.4778	0.4629	0.4838
6	0.9690	0.4505	0.3993	0.6633	0.4985	0.4835	0.5002
7	0.9681	0.4907	0.3978	0.6370	0.4898	0.4742	0.4882
8	0.9685	0.4707	0.3871	0.6551	0.4866	0.4715	0.4890
9	0.9680	0.3833	0.3864	0.6410	0.4822	0.4667	0.4827
Mean	0.9685	0.4535	0.3891	0.6543	0.4879	0.4728	0.4899
Std	0.0003	0.0364	0.0092	0.0088	0.0073	0.0074	0.0065

```
In [ ]: predict_model(rf_model)
```

	Model	Accuracy	AUC	Recall	Prec.	F1	Kappa	MCC
0	Random Forest Classifier	0.9700	0.9000	0.4280	0.6731	0.5233	0.5085	0.5225

Out[ ]:

	Shop	Order_Date_FK	ProductCode	OriginalSaleAmountInclVAT	CustomerID	SaleDocumentNu
321657	80	20210605	-3594386829413854439	5.950000	5499616353778023976	2348
1036803	49	20210613	5078107360208977787	79.949997	4293007591427090667	2357
711575	80	20210605	8336052905460670881	99.949997	-1649441281930260175	2349
1712731	102	20210808	5804696667189589341	89.949997	-6403828227756768410	2408
1233378	80	20210708	-3191406221372595431	39.950001	-8238248260529046158	2379
...	...	...	...	...	...	
508314	63	20210730	-8086085753620229722	79.949997	-8574235684826723583	2399
1481982	102	20210623	2983231773606615865	28.950001	5112178596325179500	2370
588490	37	20210612	-4404528245931608210	75.949997	5379382321311379562	2358
400422	11	20210602	9052019780650506997	49.950001	2107955572144276411	2345
1319459	34	20210627	-4845953791492409240	6.950000	-5808390096124119920	2369

527967 rows × 14 columns

```
In [ ]: save_model(rf_model, model_name = 'random_forest_model2')
```

```
Transformation Pipeline and Model Successfully Saved
```

```
Out[ ]:  (Pipeline(memory=Memory(location=None),
                   steps=[('numerical_imputer',
                           TransformerWrapper(exclude=None,
                                              include=['Shop', 'Order_Date_FK',
                                                       'ProductCode',
                                                       'OriginalSaleAmountInclVAT',
                                                       'CustomerID', 'SaleDocumentNumber',
                                                       'RevenueInclVAT',
                                                       'CostPriceExclVAT', 'BrandName',
                                                       'ModelGroup', 'ProductGroup'],
                                              transformer=SimpleImputer(add_indicator=False,
                                                                        copy=True,
                                                                        fill_value=...
                           RandomForestClassifier(bootstrap=True, ccp_alpha=0.0,
                                                  class_weight=None, criterion='gini',
                                                  max_depth=None, max_features='sqrt',
                                                  max_leaf_nodes=None, max_samples=None,
                                                  min_impurity_decrease=0.0,
                                                  min_samples_leaf=1, min_samples_split=2,
                                                  min_weight_fraction_leaf=0.0,
                                                  monotonic_cst=None, n_estimators=100,
                                                  n_jobs=-1, oob_score=False,
                                                  random_state=123, verbose=0,
                                                  warm_start=False))],
                   verbose=False),
          'random_forest_model2.pkl')
```