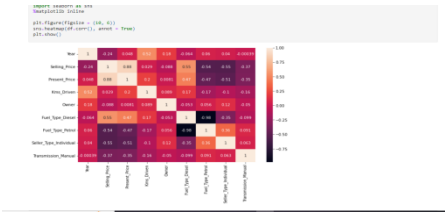
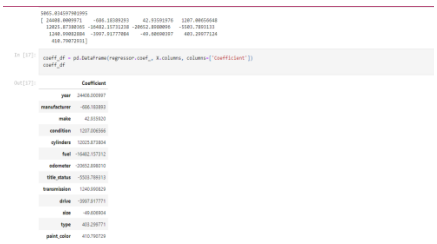
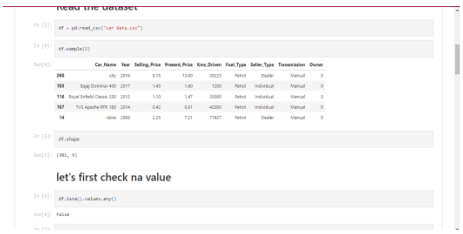
Project Development Phase
Model Performance Test

Date	10 November 2022
Team ID	PNT2022TMID13743
Project Name	Project – Car Resale Value Prediction Using machine learning
Maximum Marks	10 Marks

Model Performance Testing:

S.No.	Parameter	Values	Screenshot

1.	Metrics	**Classification Model:**
		# Linear regression classifier from from sklearn.model_selection import train_test_split X_train, X_test, y_train, y_test = train_test_split(X, y, test_size = 0.20)
		model.score(X_test, y_test))
		Accuracy: accuracy_score(y_test, y_pred_csv) #92%
		Classification Report :
		print(classification_report(y_test, y_pred_csv))
		Confusion Matrix:
		cm = confusion_matrix(y_test, y_pred_csv)
		plt.title('Heatmap of Confusion Matrix', fontsize = 12)
		sns.heatmap(cm, annot = True, fmt = "d")
		plt.show()
		#Random Forest model:
		from sklearn.model_selection
		X_pred = np.zeros_like(X.columns)
		feature_list = [year, present_price, kms, owner, diesel, petrol, individual, manual]
		Accuracy:
		predict_price(7, 9.54, 43000, 0, 1, 0, 0, 1, "sx4")
			
			 
			 
			 

		Classification Report : print(classification_report(y_test, y_pred))	

2.	Tune the Model	**Hyperparameter Tuning :**
		param": { "n_estimators": [int(x) for x in np.linspace(100, 1200, 12)],
		"max_features": ["auto", "sqrt"],
		"min_samples_split": [2, 5, 10, 15, 100],
		"min_samples_leaf": [1, 2, 5, 10] }
		#Randomize Search cv for hyperparamter tuninig from sklearn.model_selection import RandomizedSearchCV
		best_models
		model_tunning.fit(X_train, y_train)
		best_models[model_name] = model_tunning
		# print how our model looks after hyper-parameter tuning
		print(name, " : ", model.score(X_test, y_test))