Question 1

What is the optimal value of alpha for ridge and lasso regression? What will be the changes in the model if you choose double the value of alpha for both ridge and lasso? What will be the most important predictor variables after the change is implemented?

Ridge Alpha 1 and Lasso Alpha 10

```
ridge2 = Ridge(alpha=alpha)
ridge2.fit(X_train, y_train)
```

```
Ridge(alpha=1)
```

```
y_pred_train = ridge2.predict(X_train)
y_pred_test = ridge2.predict(X_test)

metric2 = []
r2_train_lr = r2_score(y_train, y_pred_train)
print(r2_train_lr)
metric2.append(r2_train_lr)

r2_test_lr = r2_score(y_test, y_pred_test)
print(r2_test_lr)
metric2.append(r2_test_lr)

rss1_lr = np.sum(np.square(y_train - y_pred_train))
print(rss1_lr)
metric2.append(rss1_lr)

rss2_lr = np.sum(np.square(y_test - y_pred_test))
print(rss2_lr)
metric2.append(rss2_lr)

mse_train_lr = mean_squared_error(y_train, y_pred_train)
print(mse_train_lr)
metric2.append(mse_train_lr**0.5)

mse_test_lr = mean_squared_error(y_test, y_pred_test)
print(mse_test_lr)
metric2.append(mse_test_lr**0.5)
```

For Alpha 1:

```
0.9131124037305709
0.8675909614077975
87.58269703958456
56.22760212435563
0.08688759626942913
0.12985589405162962
```

For Alpha 2:

```
0.9095438627670711
0.8721819425808334
91.1797863307923
54.278038367209916
0.09045613723292886
0.12535343733766724
```

For Alpha 3:

```
0.906532918814245
0.8747552566958273
94.21481783524099
53.18527850929347
0.093467081185754 95
0.1228297425156893
```

R2score on training data has decreased but it has increased on testing data.

```
lasso20 = Lasso(alpha=alpha)
lasso20.fit(X_train, y_train)
```

```
Lasso(alpha=10)
```

```
# Lets calculate some metrics such as R2 score, RSS and RM
y_pred_train = lasso20.predict(X_train)
y_pred_test = lasso20.predict(X_test)

metric3 = []
r2_train_lr = r2_score(y_train, y_pred_train)
print(r2_train_lr)
metric3.append(r2_train_lr)

r2_test_lr = r2_score(y_test, y_pred_test)
print(r2_test_lr)
metric3.append(r2_test_lr)

rss1_lr = np.sum(np.square(y_train - y_pred_train))
print(rss1_lr)
metric3.append(rss1_lr)

rss2_lr = np.sum(np.square(y_test - y_pred_test))
print(rss2_lr)
metric3.append(rss2_lr)

mse_train_lr = mean_squared_error(y_train, y_pred_train)
print(mse_train_lr)
metric3.append(mse_train_lr**0.5)

mse_test_lr = mean_squared_error(y_test, y_pred_test)
print(mse_test_lr)
metric3.append(mse_test_lr**0.5)
```

Changed value of Lasso from 10 to 20

For lasso = 10

R2score of training data has decrease and it has increase on testing data

Question 2

You have determined the optimal value of lambda for ridge and lasso regression during the assignment. Now, which one will you choose to apply and why?

The r2_score of lasso is slightly higher than lasso for the test dataset so we will choose lasso regression to solve this problem

Question 3

After building the model, you realised that the five most important predictor variables in the lasso model are not available in the incoming data. You will now have to create another model excluding the five most important predictor variables. Which are the five most important predictor variables now?

```
In [199]: X_train.columns

Out[199]: Index(['OverallQual', 'OverallCond', 'MasVnrArea', 'BsmtQual', 'BsmtExposure',
                 'BsmtFinSF1', 'BsmtFinSF2', 'BsmtUnfSF', 'TotalBsmtSF', '1stFlrSF',
                 '2ndFlrSF', 'LowQualFinSF', 'GrLivArea', 'BedroomAbvGr', 'KitchenAbvGr',
                 'KitchenQual', 'TotRmsAbvGrd', 'GarageArea', 'MSZoning_FV',
                 'MSZoning_RH', 'MSZoning_RL', 'MSZoning_RM', 'LandContour_Low',
                 'LotConfig_CulDSac', 'LotConfig_FR3', 'Neighborhood_Blueste',
                 'Neighborhood_BrDale', 'Neighborhood_BrkSide', 'Neighborhood_Crawfor',
                 'Neighborhood_NoRidge', 'Neighborhood_NridgHt', 'Neighborhood_Somerst',
                 'Neighborhood_StoneBr', 'Condition1_Norm', 'Condition2_PosN',
                 'BldgType_Twnhs', 'BldgType_TwnhsE', 'HouseStyle_1Story',
                 'HouseStyle_2.5Fin', 'HouseStyle_2.5Unf', 'RoofStyle_Gable',
                 'RoofStyle_Gambrel', 'RoofStyle_Shed', 'RoofMatl_Membran',
                 'RoofMatl_Metal', 'RoofMatl_Roll', 'RoofMatl_Tar&Grv',
                 'RoofMatl_WdShngl', 'Exterior1st_BrkFace', 'Exterior1st_CBlock',
                 'Exterior1st_Wd Sdng', 'Exterior2nd_Brk Cmn', 'Exterior2nd_CBlock',
                 'Exterior2nd_Other', 'Exterior2nd_Wd Sdng', 'MasVnrType_BrkFace',
                 'MasVnrType_None', 'MasVnrType_Stone', 'Foundation_PConc',
                 'Foundation_Slab', 'Heating_Wall', 'Functional_Mod', 'Functional_Sev',
                 'Functional_Typ', 'SaleType_ConLD', 'SaleType_ConLI', 'SaleType_New',
                 'SaleType_Oth', 'SaleType_WD', 'SaleCondition_Partial'],
                dtype='object')
```

'OverallQual', 'OverallCond', 'MasVnrArea', 'BsmtQual', 'BsmtExposure' Five most important columns

Question 4

How can you make sure that a model is robust and generalisable? What are the implications of the same for the accuracy of the model and why?

The model should be generalized so that the test accuracy is not lesser than the training score. The model should be accurate for datasets other than the ones which were used during training. Too much importance should not given to the outliers so that the accuracy predicted by the model is high. To ensure that this is not the case, the outliers analysis needs to be done and only those which are relevant

to the dataset need to be retained. Those outliers which it does not make sense to keep must be removed from the dataset. If the model is not robust, It cannot be trusted for predictive analysis.