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 to double the value of alpha for both ridge and lasso? What will be the most important predictor variables after the change is implemented?

Ans: The best alpha from best_param value is 10 however I choose to use alpha as 2 .because of the following explanation.

ridge regression:

When I plot the curve between negative mean absolute error and alpha we see that as the value of alpha increase from 0 the error term decreases, and the test error is showing increasing trend when value of alpha increases .when the value of alpha is 2 the test error is minimum, so we decided to go

lasso regression:

best param value 0.001

I have decided to keep very small value that is 0.01, when we increase the value of alpha the model try to penalize more and try to make most of the coefficient value zero. will take the value of alpha equal to 10 the model will apply more penalty on the curve and try to make the model more generalized that is making model more complex and the model is now thinking to fit every data of the data set .from the graph we can see that when alpha is 10 we get more error for both test and train.

Similarly, when we increase the value of alpha for lasso we try to penalize more our model and more coefficient of the variable will reduced to zero, when we increase the value of our r2 square also decreases.

with alpha 2

```
# ridge regression
lm = Ridge(alpha=2)
lm.fit(X_train, y_train)

# predict
y_train_pred = lm.predict(X_train)
print(metrics.r2_score(y_true=y_train, y_pred=y_trai
y_test_pred = lm.predict(X_test)
print(metrics.r2_score(y_true=y_test, y_pred=y_test_

0.9364594823911133
0.907759707946659
```

Alpha 10

```
# ridge regression
lm = Ridge(alpha=10)
lm.fit(X_train, y_train)

# predict
y_train_pred = lm.predict(X_train)
print(metrics.r2_score(y_true=y_train, y_pred=y_train_pred))
y_test_pred = lm.predict(X_test)
print(metrics.r2_score(y_true=y_test, y_pred=y_test_pred))
```

```
0.9308424818872958
0.9096659084948928
```

Slight decrease in train data and increase in test data

The most important variable after the changes has been implemented for ridge regression are as follows:-

1. MSZoning_FV
2. MSZoning_RL
3. Neighborhood_Crawfor
4. MSZoning_RH
5. MSZoning_RM

The most important variable after the changes has been implemented for lasso regression are as follows:- 1. GrLivArea

2. OverallQual
3. OverallCond
4. TotalBsmtSF
5. BsmtFinSF1

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?

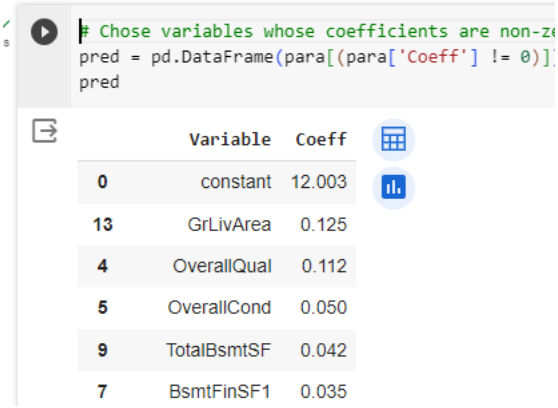
Ans: will select lasso regression because it creates a simple model by making insignificant variables to 0 and creating a simple model.

It is important to regularize coefficients and improve the prediction accuracy also with the decrease in variance, and making the model interpretably. Ridge regression includes all variables in final model unlike Lasso Regression. Lasso regression, uses a tuning parameter called lambda as the penalty is absolute value of magnitude of coefficients which is identified by cross validation. As the lambda value increases Lasso shrinks the coefficient towards zero and it make the variables exactly equal to 0. Lasso also does variable selection. When lambda value is small it performs simple linear regression and as lambda value increases, shrinkage takes place and variables with 0 value are neglected by the model.

Question 3

After building the model, you realized 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?

Ans: The 5 most important variables are ignore the constant row 0



```
# Chose variables whose coefficients are non-ze
pred = pd.DataFrame(para[(para['Coeff'] != 0)]
pred
```

	Variable	Coeff
0	constant	12.003
13	GrLivArea	0.125
4	OverallQual	0.112
5	OverallCond	0.050
9	TotalBsmtSF	0.042
7	BsmtFinSF1	0.035

Question 4

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

Ans: there should be tradeoff between Bias and Variance, we need to select a model which is optimum . that is thought the accuracy decreases the variance should not be high . The simpler the model the more the bias but less variance and more generalizable. Model will perform equally well on both training and test data i.e. the accuracy does not change much for training and test data.

Bias: Bias is error in model, when the model is weak to learn from the data. High bias means model is unable to learn details in the data. Model performs poor on training and testing data.

Variance: Variance is error in model, when model tries to over learn from the data. High variance means model performs exceptionally well on training data as it has very well trained on this of data but performs very poor on testing data as it was unseen data for the model. It is important to have balance in Bias and Variance to avoid overfitting and under-fitting of data