Question 1

Answer: 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?

Regularisation is done to keep a check on the complexity of the model. A model like this with a lot of features (variables can tend be overly complex which leads to decrease in bias and very high variance which is not acceptable for a stable model) To keep the complexity in check we need to regularise the models by using Ridge or Lasso. By doing so we penalise the model for its complexity.

The penalty term also known as Lambda (Alpha in SciKit_Learn) is the coefficient of regularisation. The final values of alpha for both Ridge and Lasso were as follows

Ridge Alpha = 100

Lasso Alpha =0.002

For the final Ridge and lasso model the final values of R2 and MSE are compared with the models with Doubled value of alpha.

Metric	**Ridge**		**Lasso**	
Alpha	100	200	0.002	0.004
R2 Train	89.65	88.66	88.86	86.78
R2 Test	87.89	87.29	87.05	86.86
MSE	0.0199	0.020911	0.021308	0.021614

For Ridge, on Doubling the Alpha the difference between the R2 scores of train and test decreases but also the MSE increases which can be said as minimal and hence for the case of Ridge it turned out to be a good idea

For Lasso, on Doubling the Alpha the difference between the R2 scores of train and test decreases but the values change, the R2 of test becomes greater than that of the train data which is not good and hence is not a good model

The values of their coefficients of Constant and other features, also change when their Alpha is doubled. But their MSE do not undergo major changes.

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?

Answer: For the given Data set the optimal values of Lambda (Alpha) for both Ridge and Lasso were determined on the basis of the difference between their error terms and R2 values between the train and test data.

Metric	**Ridge**	**Lasso**
Alpha	100	0.002
R2 Train	89.65	88.86
R2 Test	87.89	87.05
MSE	0.0199	0.021308
Feature count	260	42

For a similar amount of MSE (approx. 0.02) the difference between the R2 of train and test models of Ridge was more than that of Lasso, also for the same error the number of variables(features) used by Ridge was 260 whereas lasso used only 42 features which was way less complex which is more desirable and practical real-world applications.

Due to all of the above-mentioned reasons The Lasso model was selected over Ridge model for further analysis.

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?

Answer: For the given Data set the five most important variables for the Lasso model are as follows

The top 5 features of the Lasso model originally were:

1. OverallQual
2. Neighborhood_Crawfor
3. TotRmsAbvGrd
4. GarageArea
5. Condition1_Norm

	Feature	Value
0	constant	11.994
4	OverallQual	0.118
26	Neighborhood_Crawfor	0.070
13	TotRmsAbvGrd	0.053
15	GarageArea	0.048
31	Condition1_Norm	0.043

After removing the top 5 features of the original Lasso model the new top 5 features are:

1. Neighborhood_NridgHt
2. 1stFlrSF
3. FullBath
4. HalfBath
5. Foundation_PConc

	Feature	Value
0	constant	11.955
25	Neighborhood_NridgHt	0.128
6	1stFlrSF	0.085
8	FullBath	0.075
9	HalfBath	0.070
32	Foundation_PConc	0.068

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?

Answer: For a machine learning model to be selected for use after tuning the most important factor is the general usage of the model to predict based on new data for a long period of time, this is known as generalisability and robustness of the model.

The model should learn from the important trends of the training data and not the datapoint itself, this will lead to overfitting.

The amount of data required to train the model should not be very large, this can hinder the time required to deploy the model and can be rendered useless for small scale testing and small-scale application with very small amount of data to train.

For Robust ness the model should have low variance and high bias. This cannot be provided by an overly complex model.

Based on all of the above condition it can be clearly said that a model to be robust and generalisable should be, as per Occam's razer, simple for use but not too simple.

In the simplest of terms, a model should be as simple as necessary.