1) For the entire dataset, please perform the data cleaning as instructed before; namely, exclude the variables PassengerID, Name, Ticket, and Cabin and delete missing values in the variable Age. Please report how many passengers are left after this step.

```
library(dplyr)
```

```
##
## Attaching package: 'dplyr'
```

```
## The following objects are masked from 'package:stats':
##
##     filter, lag
```

```
## The following objects are masked from 'package:base':
##
##     intersect, setdiff, setequal, union
```

```
titanic_data <- read.csv("Titanic2.csv")

titanic_cleaned <- titanic_data %>%
  select(-PassengerId, -Name, -Ticket, -Cabin)

titanic_cleaned <- titanic_cleaned %>%
  filter(!is.na(Age))

n_passengers <- nrow(titanic_cleaned)
cat("Number of passengers left after cleaning:", n_passengers, "\n")
```

```
## Number of passengers left after cleaning: 714
```

```
str(titanic_cleaned)
```

```
## 'data.frame':    714 obs. of  8 variables:
##  $ Survived: int  0 1 1 1 0 0 0 1 1 1 ...
##  $ Pclass  : int  3 1 3 1 3 1 3 3 2 3 ...
##  $ Sex     : chr  "male" "female" "female" "female" ...
##  $ Age     : num  22 38 26 35 35 54 2 27 14 4 ...
##  $ SibSp   : int  1 1 0 1 0 0 3 0 1 1 ...
##  $ Parch   : int  0 0 0 0 0 0 1 2 0 1 ...
##  $ Fare    : num  7.25 71.28 7.92 53.1 8.05 ...
##  $ Embarked: chr  "S" "C" "S" "S" ...
```

2) Please use the "str()" function to examine the variable type in our dataset. Please note that R will recognize a character-valued function as a categorical variable (namely a Factor variable) automatically and therefore you do not need to transform this type of variable. However, you do need to recode the numerical-valued categorical variables using the "as.factor()" function so that R can recognize these as categorical variables. Now, please recode the "Survived" and the "Pclass" variables as factors.

```
titanic_cleaned <- titanic_cleaned %>%
  mutate(Survived = as.factor(Survived),
         Pclass = as.factor(Pclass))

str(titanic_cleaned)
```

```
## 'data.frame':    714 obs. of  8 variables:
##  $ Survived: Factor w/ 2 levels "0","1": 1 2 2 2 1 1 1 2 2 2 ...
##  $ Pclass  : Factor w/ 3 levels "1","2","3": 3 1 3 1 3 1 3 3 2 3 ...
##  $ Sex     : chr  "male" "female" "female" "female" ...
```

```
##  $ Age     : num  22 38 26 35 35 54 2 27 14 4 ...
##  $ SibSp   : int  1 1 0 1 0 0 3 0 1 1 ...
##  $ Parch   : int  0 0 0 0 0 0 1 2 0 1 ...
##  $ Fare    : num  7.25 71.28 7.92 53.1 8.05 ...
##  $ Embarked: chr  "S" "C" "S" "S" ...
```

3) Now please use the random seed 123 to divide the cleaned data into 80% training and 20% testing.

```
set.seed(123)

train_index <- sample(1:nrow(titanic_cleaned), 0.8 * nrow(titanic_cleaned))
train_data <- titanic_cleaned[train_index, ]
test_data <- titanic_cleaned[-train_index, ]
```

4) Then you shall fit a logistic regression model with all the other 7 predictors using the training data.

```
logistic_model <- glm(Survived ~ ., data = train_data, family = binomial)

summary(logistic_model)
```

```
##
## Call:
## glm(formula = Survived ~ ., family = binomial, data = train_data)
##
## Coefficients:
##              Estimate Std. Error z value Pr(>|z|)
## (Intercept)  4.040623   0.573169   7.050 1.79e-12 ***
## Pclass2     -1.090644   0.363805  -2.998  0.00272 **
## Pclass3     -2.212574   0.367180  -6.026 1.68e-09 ***
## Sexmale     -2.563812   0.242178 -10.586  < 2e-16 ***
## Age         -0.038177   0.008893  -4.293 1.76e-05 ***
## SibSp       -0.331146   0.139801  -2.369  0.01785 *
## Parch       -0.073210   0.130597  -0.561  0.57509
## Fare         0.001500   0.002645   0.567  0.57060
## EmbarkedQ   -0.633477   0.629029  -1.007  0.31390
## EmbarkedS   -0.391929   0.299565  -1.308  0.19076
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 766.59  on 570  degrees of freedom
## Residual deviance: 519.49  on 561  degrees of freedom
## AIC: 539.49
##
## Number of Fisher Scoring iterations: 5
```

5) Please use this fitted model based on the training data to predict the response variable "Survived" (whether the subject survived or not) for the testing data. Please generate the confusion matrix, and report:

The overall accuracy;

The sensitivity (that is, the probability a subject is predicted to have survived given that he/she had survived);

The specificity (that is, the probability a subject is predicted to have not survived given that he/she had not survived).

```r
test_predictions <- predict(logistic_model, test_data, type = "response")
test_predictions <- ifelse(test_predictions > 0.5, 1, 0)

test_predictions <- as.factor(test_predictions)

confusion_matrix <- table(Predicted = test_predictions, Actual = test_data$Survived)
confusion_matrix
```

```
##          Actual
## Predicted  0  1
##         0 73 18
##         1  6 46
```

```r
accuracy <- sum(diag(confusion_matrix)) / sum(confusion_matrix)
cat("Overall Accuracy:", accuracy, "\n")
```

```
## Overall Accuracy: 0.8321678
```

```r
sensitivity <- confusion_matrix[2, 2] / sum(confusion_matrix[, 2])
cat("Sensitivity:", sensitivity, "\n")
```

```
## Sensitivity: 0.71875
```

```r
specificity <- confusion_matrix[1, 1] / sum(confusion_matrix[, 1])
cat("Specificity:", specificity, "\n")
```

```
## Specificity: 0.9240506
```

6) Now we have recovered the record of additional passengers as follows. Please predict whether these passengers have survived or not.

```r
additional_passengers <- data.frame(
  PassengerId = c(892, 893, 894),
  Pclass = as.factor(c(3, 1, 2)),
  Sex = c("male", "female", "male"),
  Age = c(24, 68, 41),
  SibSp = c(1, 0, 1),
  Parch = c(0, 0, 2),
  Fare = c(8.42, 24.34, 41.93),
  Embarked = c("Q", "C", "S")
)

additional_predictions <- predict(logistic_model, additional_passengers, type = "response")
additional_predictions <- ifelse(additional_predictions > 0.5, "Survived", "Not Survived")

results <- data.frame(
  PassengerId = additional_passengers$PassengerId,
  Survival_Prediction = additional_predictions
)

print(results)
```

```
##   PassengerId Survival_Prediction
## 1         892        Not Survived
## 2         893            Survived
## 3         894        Not Survived
```