Summary Report

1. Which Features Were Most Important and Why?

After training and tuning the Random Forest Classifier, we extracted the top 10 most important features using feature_importances_. The top features were:

1. Fare
2. Age
3. Fare_Per_Person
4. Family_Size
5. poly_Fare^2
6. poly_Age
7. poly_Age Fare
8. poly_Fare
9. Age_Fare
10. Age_Group_Code

These features were important because:

- Fare and Fare_Per_Person reflect socio-economic status, which influenced survival (e.g., wealthier passengers had better access to lifeboats).
- Age is important because children are more likely to be saved.
- Family_Size captures the number of family members onboard; very large or very small families had different survival patterns.
- Polynomial features like poly_Fare^2 and poly_Age Fare helped capture non-linear relationships between features and survival.

2. How Hyperparameter Tuning Improved Performance

We used GridSearchCV to test combinations of the following hyperparameters:

```
param_grid = {
    'n_estimators': [50, 100, 200],
    'max_depth': [3, 5, 10, None],
    'min_samples_split': [2, 5, 10]
}
```

This helped the model avoid underfitting (too simple) and overfitting (too complex).

Best Parameters Found:

```
✅ Best Hyperparameters Found:
{'max_depth': 3, 'min_samples_split': 2, 'n_estimators': 100}
```

This combination gave a good balance of depth and number of trees, leading to improved generalization on the test set.

3. What Was the Final Test Accuracy?

After using the best estimator from the grid search, we evaluated it on the 20% test set:

```
📊 Test Set Evaluation with Tuned Random Forest:
Accuracy:  0.6983
Precision: 0.7778
Recall:    0.3784
F1 Score:  0.5091
```

This shows that the model performs reasonably well and captures the pattern of survival effectively.

4. One Thing That Surprised Me

One surprising insight was how much feature engineering (e.g., Fare_Per_Person, polynomial features) improved the model's performance.

Also, the learning curve showed that:

- Training accuracy was very high with small data, but dropped and stabilized as the size increased.
- Validation accuracy improved with more training data but never matched training accuracy.

This indicated some overfitting, but also suggested that adding more diverse data or reducing complexity could further help.