In [1]:
```
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
from sklearn.model_selection import train_test_split
from sklearn.linear_model import LinearRegression
from sklearn.metrics import mean_squared_error, r2_score
```

In [2]:
```
df = pd.read_csv("StudentsPerformance.csv")
df.head()
```

Out[2]:

	gender	race/ethnicity	parental level of education	lunch	test preparation course	math score	reading score	writing score
0	female	group B	bachelor's degree	standard	none	72	72	74
1	female	group C	some college	standard	completed	69	90	88
2	female	group B	master's degree	standard	none	90	95	93
3	male	group A	associate's degree	free/reduced	none	47	57	44
4	male	group C	some college	standard	none	76	78	75

In [3]:
```
df.columns
```

Out[3]:
```
Index(['gender', 'race/ethnicity', 'parental level of education', 'lunch',
       'test preparation course', 'math score', 'reading score',
       'writing score'],
      dtype='object')
```

In [6]:
```
print(df.isnull().sum())


df = df.dropna(subset=['reading score'])
categorical_columns = ['gender', 'race/ethnicity', 'parental level of education', 'lunch', 'test preparation co
df = pd.get_dummies(df, columns=categorical_columns, drop_first=True)
```

```
gender                         0
race/ethnicity                 0
parental level of education    0
lunch                          0
test preparation course        0
math score                     0
reading score                  0
writing score                  0
dtype: int64
```

In [8]:
```
X = df.drop(columns=['reading score'])
y = df['reading score']
```

In [9]:
```
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2, random_state=42)
```

In [10]:
```
model = LinearRegression()
model.fit(X_train, y_train)
```

Out[10]:
```
▾ LinearRegression
LinearRegression()
```

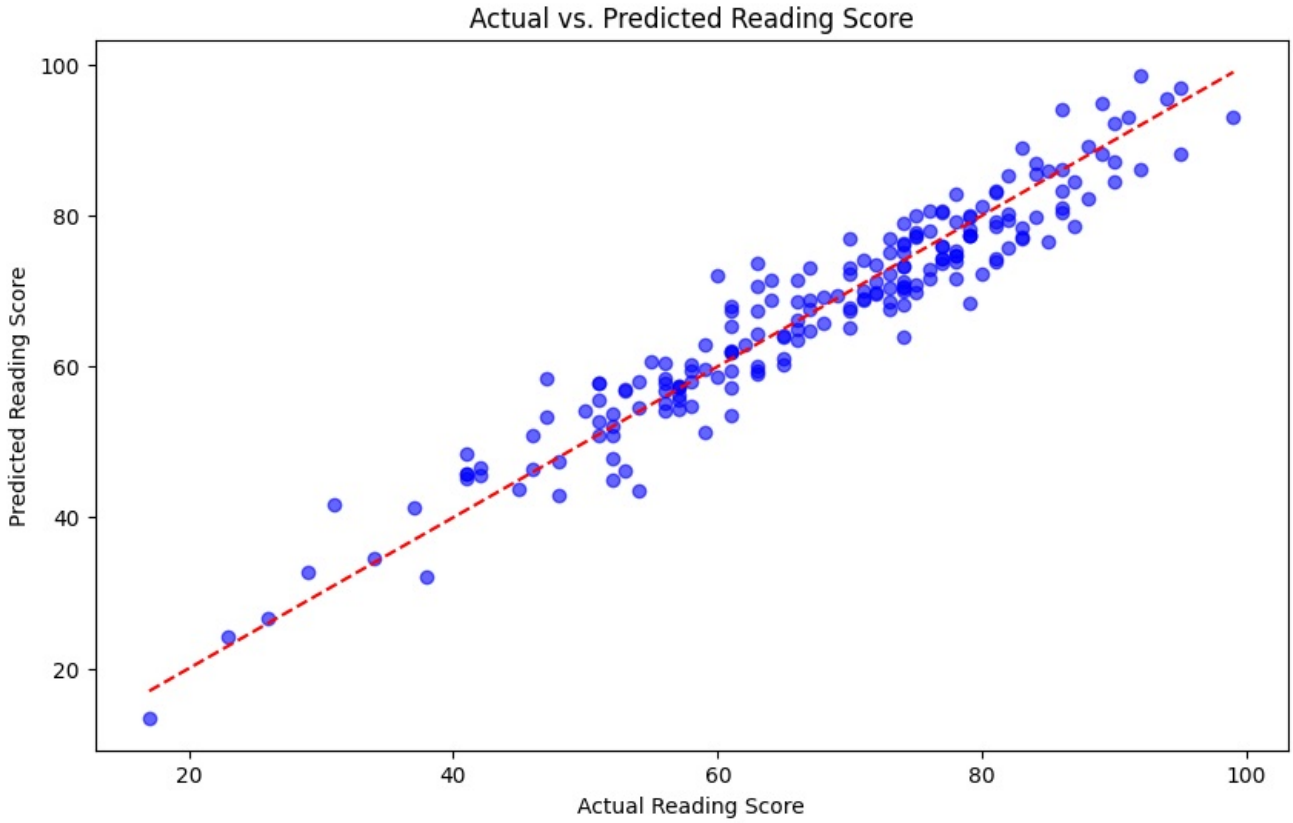
In [11]:
```
y_pred = model.predict(X_test)


mse = mean_squared_error(y_test, y_pred)
rmse = np.sqrt(mse)
r2 = r2_score(y_test, y_pred)

print(f'Mean Squared Error: {mse}')
print(f'Root Mean Squared Error: {rmse}')
print(f'R^2 Score: {r2}')
```

```
Mean Squared Error: 18.565599730519747
Root Mean Squared Error: 4.308781699102398
R^2 Score: 0.9179545661050625
```

In [12]:
```
plt.figure(figsize=(10, 6))
plt.scatter(y_test, y_pred, alpha=0.6, color='b')
plt.plot([y_test.min(), y_test.max()], [y_test.min(), y_test.max()], 'r--')  # Perfect prediction line
plt.xlabel('Actual Reading Score')
plt.ylabel('Predicted Reading Score')
plt.title('Actual vs. Predicted Reading Score')
plt.show()
```



In [ ]: