Streamlit\Comparison_of_all_regs_emp_salary.py

```python
import streamlit as st
import pickle
import numpy as np

# Load the pickled models
with open(r'C:\Users\Jan Saida\linear_regression_model.pkl', 'rb') as file:
    lin_reg = pickle.load(file)

with open(r'C:\Users\Jan Saida\polynomial_regression_model.pkl', 'rb') as file:
    lin_reg_2 = pickle.load(file)

with open(r'C:\Users\Jan Saida\svr_model.pkl', 'rb') as file:
    svr_reg = pickle.load(file)

with open(r'C:\Users\Jan Saida\knn_model.pkl', 'rb') as file:
    knn_reg = pickle.load(file)

with open(r'C:\Users\Jan Saida\decision_tree_model.pkl', 'rb') as file:
    tree_reg = pickle.load(file)

with open(r'C:\Users\Jan Saida\random_forest_model.pkl', 'rb') as file:
    forest_regressor = pickle.load(file)

with open(r'C:\Users\Jan Saida\xgboost_model.pkl', 'rb') as file:
    xgb_r = pickle.load(file)

# Polynomial Features (this is the same across different models)
from sklearn.preprocessing import PolynomialFeatures
poly_reg = PolynomialFeatures(degree=6)

# Streamlit UI
st.title('Salary Prediction using Multiple Regression Models')

# Input for Position level
position_level = st.number_input('Enter the position level (e.g., 6.5):', min_value=1.0, max_value=10.0, value=6.5)

# Polynomial Regression Prediction
position_level_poly = poly_reg.fit_transform([[position_level]])  # Apply transformation
poly_pred = lin_reg_2.predict(position_level_poly)  # Predict using the transformed features

# Linear Regression Prediction
lin_pred = lin_reg.predict([[position_level]])

# Support Vector Regression Prediction
svr_pred = svr_reg.predict([[position_level]])

# KNN Regression Prediction
knn_pred = knn_reg.predict([[position_level,0]])

# Decision Tree Prediction
tree_pred = tree_reg.predict([[position_level]])
```

```python

# Random Forest Prediction
forest_pred = forest_regressor.predict([[position_level]])

# XGBoost Prediction
xgb_pred = xgb_r.predict([[position_level]])

# Display the results
st.subheader(f'Predicted Salary for Position Level {position_level}')
st.write(f"Linear Regression Prediction: {lin_pred[0]}")
st.write(f"Polynomial Regression Prediction: {poly_pred[0]}")
st.write(f"Support Vector Regression Prediction: {svr_pred[0]}")
st.write(f"KNN Regression Prediction: {knn_pred[0]}")
st.write(f"Decision Tree Prediction: {tree_pred[0]}")
st.write(f"Random Forest Prediction: {forest_pred[0]}")
st.write(f"XGBoost Prediction: {xgb_pred[0]}")

```