Untitled4

In [1]:

```
# Import necessary libraries
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns
from sklearn.model_selection import train_test_split
from sklearn.linear_model import LinearRegression
from sklearn.metrics import mean_squared_error

# Load the dataset
url = '/Users/kiran/Desktop/insurance.csv'
data = pd.read_csv(url)

# Preview the dataset
data.head()
```

Out[1]:

	age	sex	bmi	children	smoker	region	charges
0	19	female	27.900	0	yes	southwest	16884.92400
1	18	male	33.770	1	no	southeast	1725.5230
2	28	male	33.000	3	no	southeast	4449.46200
3	33	male	22.705	0	no	northwest	21984.47061
4	32	male	28.880	0	no	northwest	3866.85520

In [3]:

```
# Check for missing values
data.isnull().sum()

# Data summary
data.describe()

# Visualize the relationships
sns.pairplot(data)
```

Out[3]:

```
<seaborn.axisgrid.PairGrid at 0x14b567800>
```

No description has been provided for this image

In [23]:

```
# Convert categorical variables using one-hot encoding
data = pd.get_dummies(data, drop_first=True)

# Define features (X) and target (y)
X = data.drop('charges', axis=1)
y = data['charges']

# Train-test split (80% training, 20% testing)
X_train, X_test, y_train, y_test = train_test_split(X, y,
test_size=0.2, random_state=42)
```

In [25]:

```
# Initialize and train the model
model = LinearRegression()
model.fit(X_train, y_train)

# Predict on the test data
y_pred = model.predict(X_test)
```

In [9]:

```
# Calculate Mean Squared Error
mse = mean_squared_error(y_test, y_pred)
print(f'Mean Squared Error: {mse}')

# Plot the results
plt.scatter(y_test, y_pred)
plt.xlabel("True Values")
plt.ylabel("Predictions")
plt.title("True Values vs Predictions")
plt.show()
```

```
Mean Squared Error: 33596915.85136147
```

No description has been provided for this image

In [ ]: