In [1]:
```
import pandas as pd
import matplotlib. pyplot as plt
import numpy as np
```

In [2]:
```
df = pd.read_csv('temperatures.csv')
```

In [3]:
```
df.isnull().sum().sum()
```

Out[3]:
```
0
```

In [4]:
```
df.shape
```

Out[4]:
```
(117, 18)
```

In [5]:
```
df.head()
```

Out[5]:

	YEAR	JAN	FEB	MAR	APR	MAY	JUN	JUL	AUG	SEP	OCT	NOV	DEC	ANNUAL	JAN-FEB	MAR-MAY	JUN-SEP	OCT-DEC
0	1901	22.40	24.14	29.07	31.91	33.41	33.18	31.21	30.39	30.47	29.97	27.31	24.49	28.96	23.27	31.46	31.27	27.25
1	1902	24.93	26.58	29.77	31.78	33.73	32.91	30.92	30.73	29.80	29.12	26.31	24.04	29.22	25.75	31.76	31.09	26.49
2	1903	23.44	25.03	27.83	31.39	32.91	33.00	31.34	29.98	29.85	29.04	26.08	23.65	28.47	24.24	30.71	30.92	26.26
3	1904	22.50	24.73	28.21	32.02	32.64	32.07	30.36	30.09	30.04	29.20	26.36	23.63	28.49	23.62	30.95	30.66	26.40
4	1905	22.00	22.83	26.68	30.01	33.32	33.25	31.44	30.68	30.12	30.67	27.52	23.82	28.30	22.25	30.00	31.33	26.57

In [6]:
```
X = df.iloc[:, [0]]
```

In [7]:
```
y = df[['JAN']]
```

In [8]:
```
from sklearn.model_selection import train_test_split
```

In [9]:
```
(X_train, X_test, y_train, y_test) = train_test_split(X, y,train_size = 0.75,test_size = 0.25)
```

In [10]:
```
from sklearn.linear_model import LinearRegression
```

In [11]:
```
lr = LinearRegression()
```

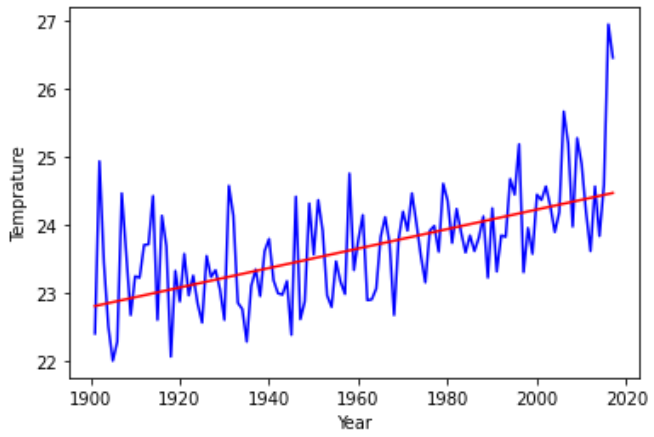
In [12]:
```
print(X_train.shape)
print(y_train.shape)
print(X_test.shape)
print(y_test.shape)
```

```
(87, 1)
(87, 1)
(30, 1)
(30, 1)
```

In [13]:
```
model = lr.fit(X_train, y_train)
```

In [14]:
```
y_pred = model.predict(X_test)
```

In [15]:
```
#plt.scatter(X_test, y_test)
plt.plot(X['YEAR'], y, c="blue")
plt.plot(X['YEAR'], model.predict(X), c="red")
plt.xlabel('Year')
plt.ylabel('Temprature')
plt.show()
```



In [16]:
```
from sklearn.metrics import r2_score, mean_squared_error, mean_absolute_error
```

In [17]:
```
print('R Square Error:', r2_score(y_test, y_pred))
print('Mean Squared Error:', mean_squared_error(y_test, y_pred))
print('Mean Absolute Error:', mean_absolute_error(y_test, y_pred))
```

```
R Square Error: 0.266545762051418
Mean Squared Error: 0.4716118084551764
Mean Absolute Error: 0.5056916454913138
```

In [18]:
```
#new model
```

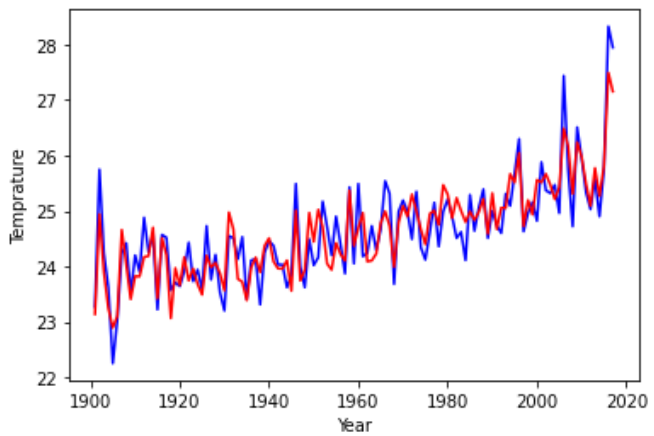
In [19]:
```
X = df.iloc[:, [0,1]]
y = df[['JAN-FEB']]
```

In [20]:
```
(X_train, X_test, y_train, y_test) = train_test_split(X, y,train_size = 0.75,test_size = 0.25)
```

In [21]:
```
model = lr.fit(X_train, y_train)
```

In [22]:
```
y_pred = model.predict(X_test)
```

In [23]:
```
plt.plot(X['YEAR'], y, c="blue")
plt.plot(X['YEAR'], model.predict(X), c="red")
plt.xlabel('Year')
plt.ylabel('Temprature')
plt.show()
```



In [24]:
```
print('R Square Error:', r2_score(y_test, y_pred))
print('Mean Squared Error:', mean_squared_error(y_test, y_pred))
print('Mean Absolute Error:', mean_absolute_error(y_test, y_pred))
```

```
R Square Error: 0.7837963571773261
Mean Squared Error: 0.19133986355866167
Mean Absolute Error: 0.3713541428530586
```

In [25]:
```
#new model
```

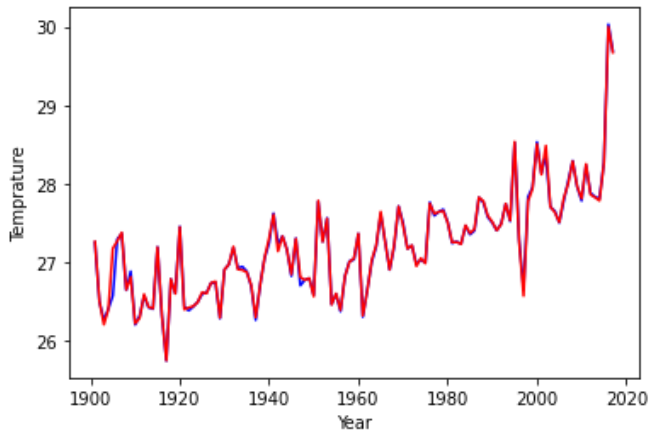
In [26]:
```
X = df.loc[:, df.columns != 'OCT-DEC']
y = df[['OCT-DEC']]
```

In [27]:
```
(X_train, X_test, y_train, y_test) = train_test_split(X, y,train_size = 0.75,test_size = 0.25)
```

In [28]:
```
model = lr.fit(X_train, y_train)
```

In [29]:
```
y_pred = model.predict(X_test)
```

In [30]:
```
plt.plot(X['YEAR'], y, c="blue")
plt.plot(X['YEAR'], model.predict(X), c="red")
plt.xlabel('Year')
plt.ylabel('Temprature')
plt.show()
```



In [31]:
```
print('R Square Error:', r2_score(y_test, y_pred))
print('Mean Squared Error:', mean_squared_error(y_test, y_pred))
print('Mean Absolute Error:', mean_absolute_error(y_test, y_pred))
```

```
R Square Error: 0.9571963587815122
Mean Squared Error: 0.012373871597806011
Mean Absolute Error: 0.0321330759834975
```