**Penjelasan Mengenai Sumber Data**

Sumber data yang dipakai dalam pengerjaan ini menggunakan data yang disediakan oleh website bernama *data.humdata.org(Humanitarian Data Exchange)* dimana website ini pelayanan atau pengerjaannya di dukung oleh OCHA(*United Nations Office for the Coordination of Humanitarian Affairs*). Sumber data ini dapat diakses di link berikut : <https://data.humdata.org/dataset/indonesia-covid-19-cases-recoveries-and-deaths-per-province> .

Pengolahan data dilakukan dengan mengambil sebagian data dari SATGAS COVID-19 yang menyajikan data dari Provinsi Jakarta, Jawa Barat, Jawa Tengah, Jawa Timur, dan provinsi lainnya di Indonesia dengan waktu dari tanggal 18 Maret 2020 hingga 29 Oktober 2020. Dalam data ini disajikan beberapa kategori seperti kasus harian dari orang yang positif COVID-19, orang yang sembuh dari COVID-19, dan orang yang Positive karena COVID-19. Data yang diberikan atau yang telah disajikan berformat *.xlsx*(*Excel Workbook*).

Data tersebut dapat diakses di link berikut :

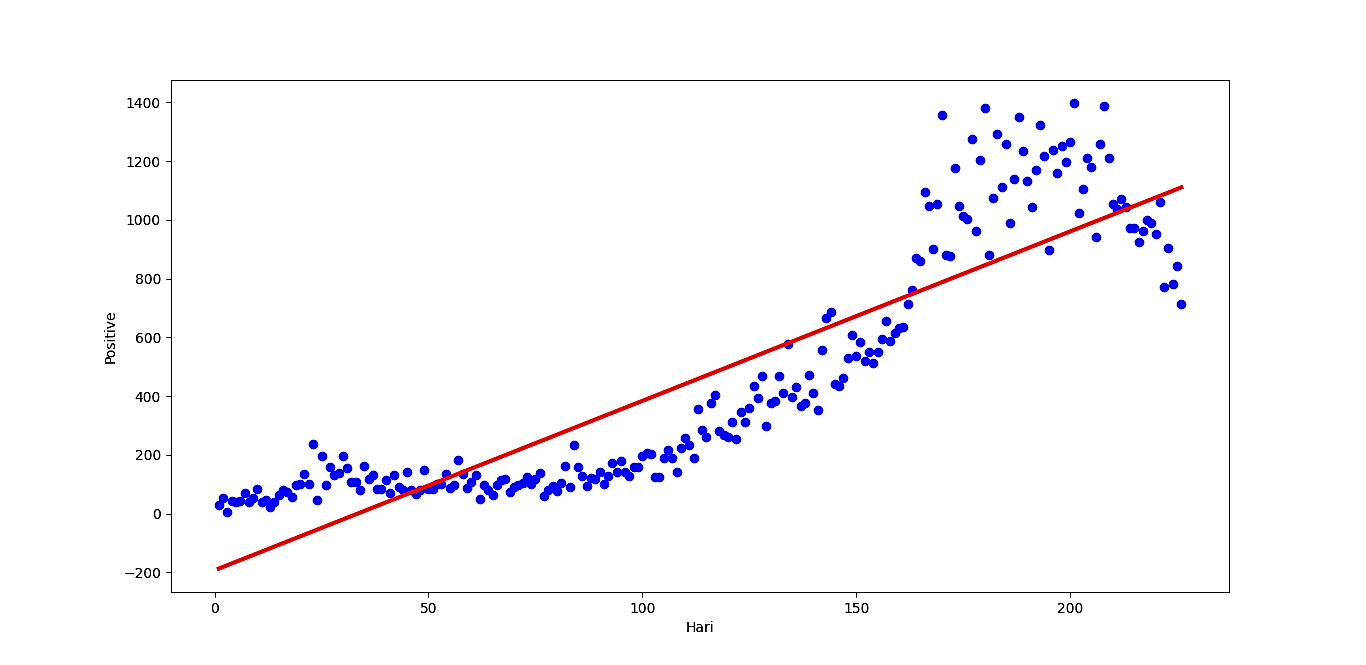
<https://drive.google.com/file/d/11wED0XH5p8OwU36qd92PqfBVacpi-EEW/view>

Selanjutnya diputuskan untuk mengambil data atau mengolah data hanya dari Provinsi Jawa Tengah dimana data yang diambil berupa hari, tanggal, kasus harian dari orang yang positif COVID-19, orang yang sembuh dari COVID-19, dan orang yang Positive karena COVID-19. Data diolah dalam tabel menggunakan excel dan seluruh data baik dari hari(Day), tanggal(Date), Hari ke-*x*(Hari), kasus harian dari orang yang positif COVID-19(Positive), orang yang sembuh dari COVID-19(Recovered), dan orang yang Positive karena COVID-19(Positive) disusun rapi dan berkesesuaian dalam satu sheet. Data yang digunakan dalam pengolahan data menggunakan data dalam kurun waktu 18 Maret – 29 Oktober 2020 dengan jumlah 226 hari.

Data ini dapat diakses di link berikut :

<https://drive.google.com/drive/folders/1UOA5Taqqnl_v0Ru7NQxvHfjGSzvvX_yY?usp=sharing>

1. **Mendapatkan persamaan fungsi linear dari data Postive**

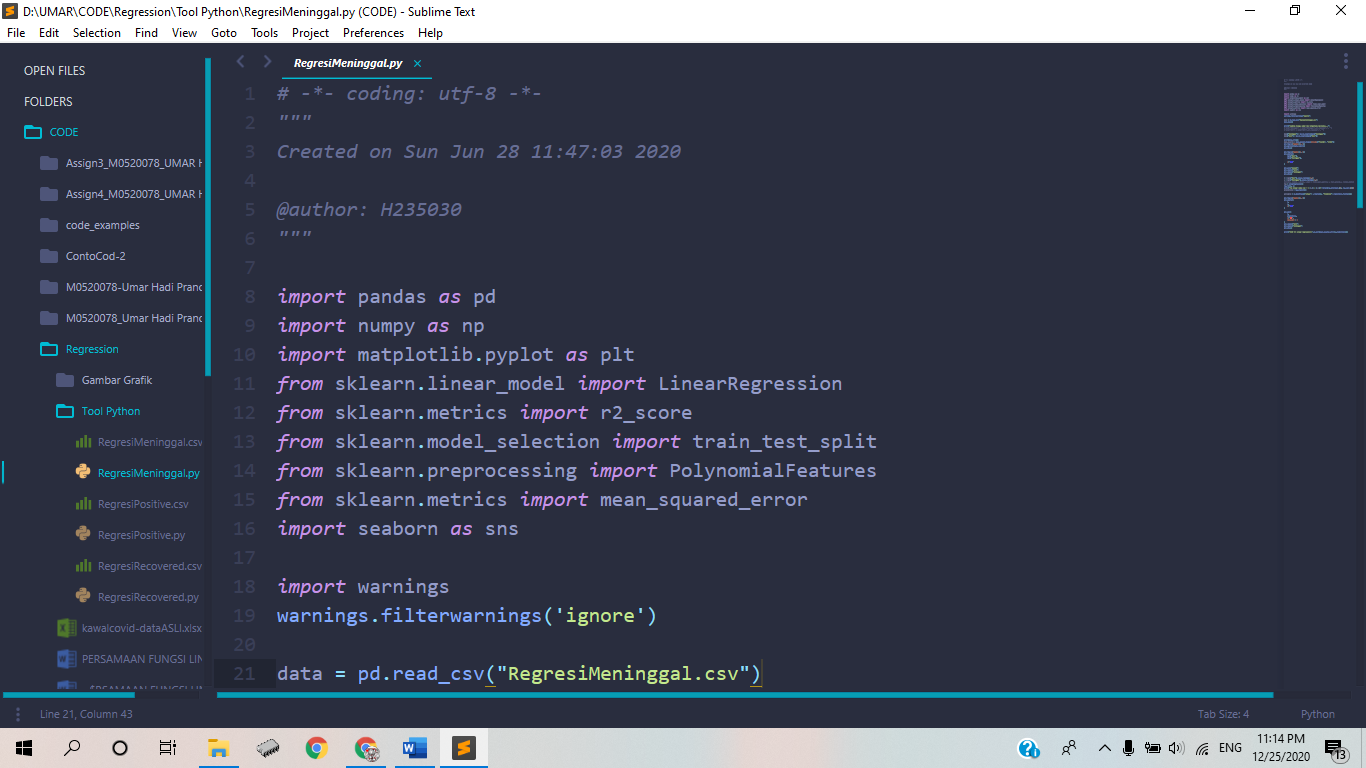


Dimana garis tersebut memiliki persamaan :

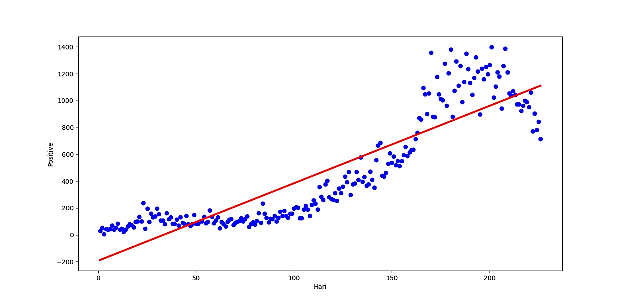
***Y = -192.97 + 5.7687X***

Pendekatan dilakukan dengan menggunakan suatu program yang dirancang atau disusun dengan menggunakan bahasa Python dengan versi 3.9.1 berikut tata cara bagaimana mendapatkan fungsi persamaan tersebut.

* 1. Program RegresionPositive,py dijalankan dengan program CMD pada windows dimana sebelumnya telah di install program atau compiler Python dan telah melakukan instalasi seaborn, numpy, OpenCV, scikit-learn dan matplotlib di dalam cmd dengan perintah pip install.
  2. Selanjutanya program akan mengimpor dahulu dengan perintah *import pandas as pd, import numpy as np, import matplotlib, pyplot as plt*, dan *import seaborn as sns*. Selanjutnya dari file-file sklearn(scikit-learn) diimpor beberapa fungsi agar dapat menjalankan suatu perintah untuk mengolah data dan menampilkan data dalam bentuk gambar dengan perintah *from sklearn.linear\_model import LinearRegression, from sklearn.metrics import r2\_score, from sklearn.model\_selection import train\_test\_split, from sklearn.preprocessing import PolynomialFeatures,* dan *from sklearn.metrics import mean\_squared\_error*

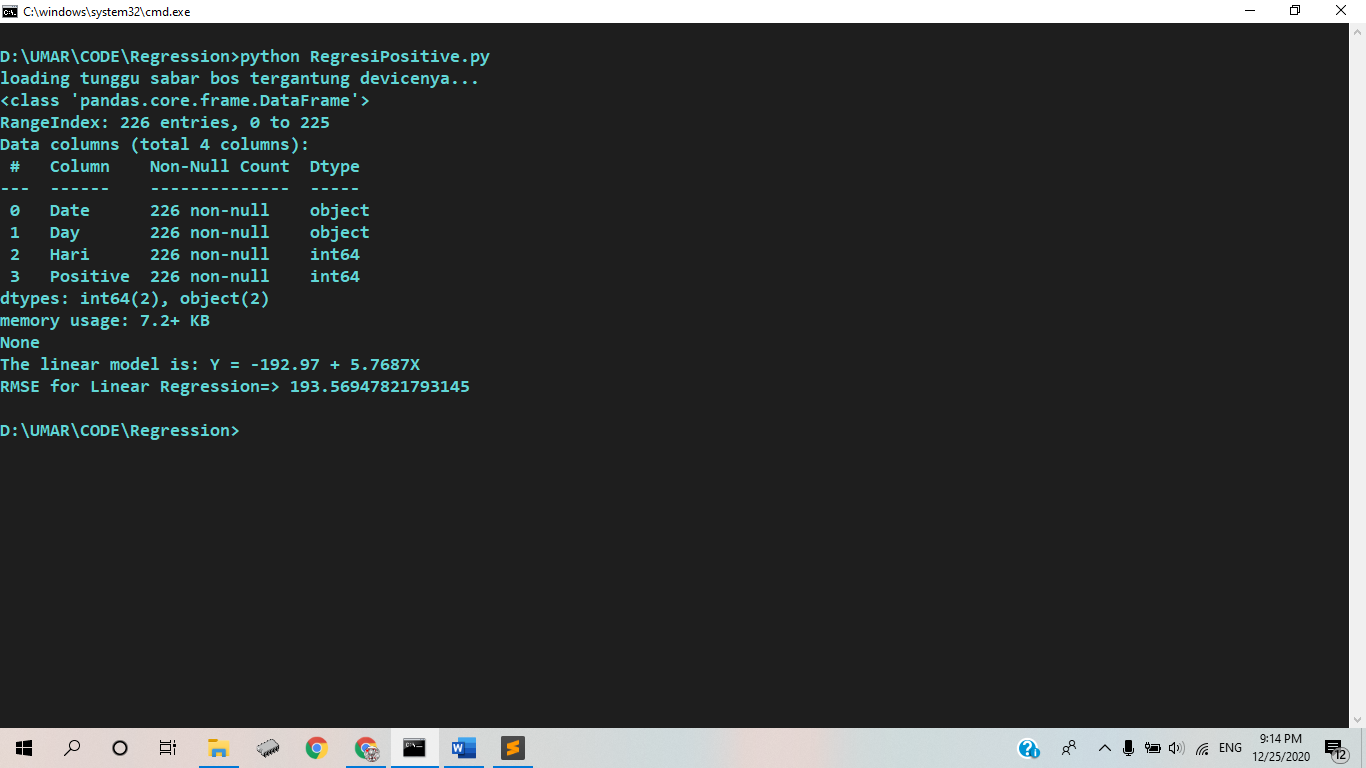


* 1. Selanjutnya akan dilakukan pembacaan data dari file “RegresiPositive.csv” dimana file ini akan dipilah dari data yang berbentuk object(tabel yang berisi string atau huruf) atau int64(tabel yang berisi angka)
  2. Selanjutnya akan dilakukan pemilihan data yang akan digunakan pada grafik dengan tipe data angka dan tidak mengandung huruf yaitu data kolom “Hari” dan data kolom “Positive” dari file .csv tersebut dengan perintah *data\_numeric = data.select\_dtypes(include=['float64', 'int64'])*, *data['Positive'] = pd.to\_numeric(data['Positive']),* dan *data['Hari'] = pd.to\_numeric(data['Hari'])*
  3. Selanjutnya akan dilakukan suatu penggambaran atau penyajian data berupa tabel grafik dengan titik X sebagai Hari dan titik Y sebagai Positive dan akan dicetak dengan warna biru dengan perintah plt.scatter *(data['Hari'], data['Positive'], c='blue')*
  4. Dan perhitungan regresi linear dilakukan dengan perintah *reg = LinearRegression() reg.fit(X, y)*. Dan ditampilkan pada program persamaan fungsi linear regresi dengan perintah *print("The linear model is: Y = {:.5} + {:.5}X".format(reg.intercept\_[0], reg.coef\_[0][0])) predictions = reg.predict(X)*
  5. Selanjutnya data disajikan dalam bentuk tabel grafik dengan menampilkan titik-titik biru(data Positive) bewarna merah dan garis merah(garis regresi linear sesuai persamaan) dengan perintah *predictions = reg.predict(X) actvspred = d.DataFrame({'Actual': y.flatten(), 'Predicted': predictions.flatten()}) plt.figure(figsize=(4, 2)) plt.scatter(X,y,c='blue' ) plt.plot X, predictions, c='red', linewidth = 3) plt.xlabel("Hari") plt.ylabel("Positive") plt.show().*
  6. Sehingga ditampilkan seperti ini,

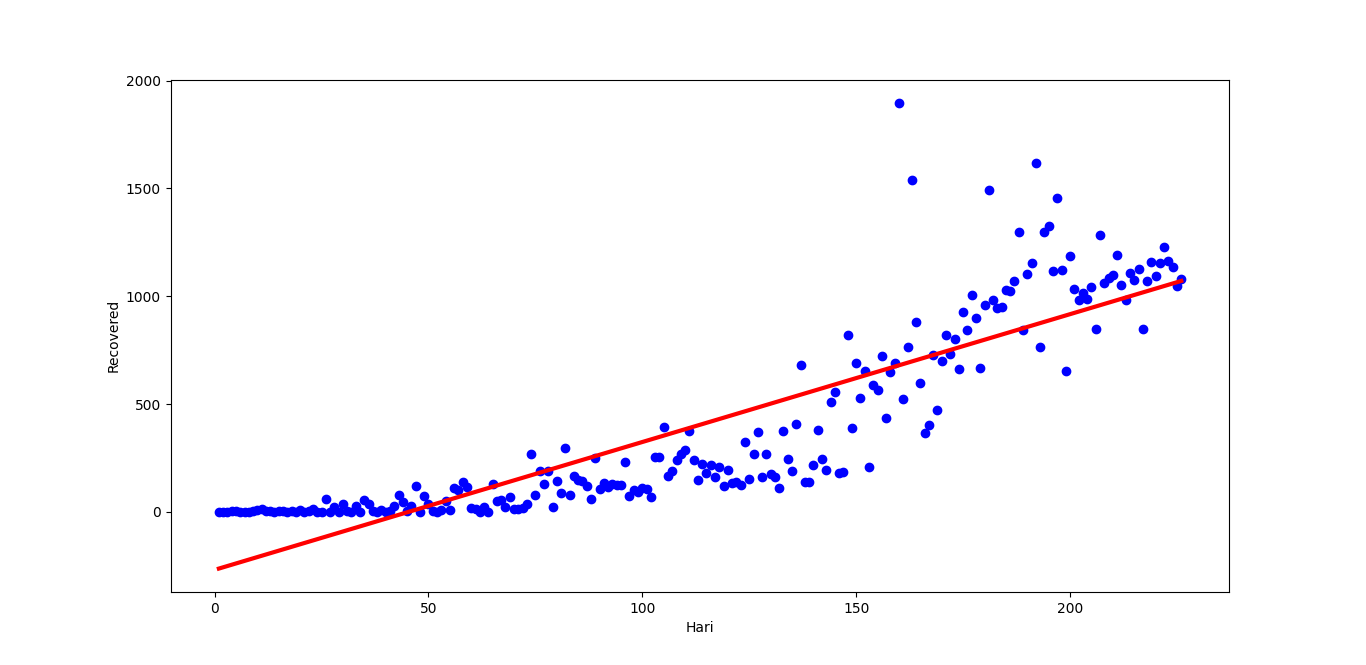


* 1. Di akhir ditampilkan Root Mean Squared Error (RMSE), yang digunakan sebagai indikator untuk mengukur kemiripan data dari titik-titik biru(data Positive) bewarna merah dan garis merah(garis regresi linear sesuai persamaan dimana merupakan data juga) dengan perintah *print('RMSE for Linear Regression=>',np.sqrt(mean\_squared\_error(y,predictions))).*

Sehingga ditampilkan seperti ini,



1. **Mendapatkan persamaan fungsi linear dari data Recovered**

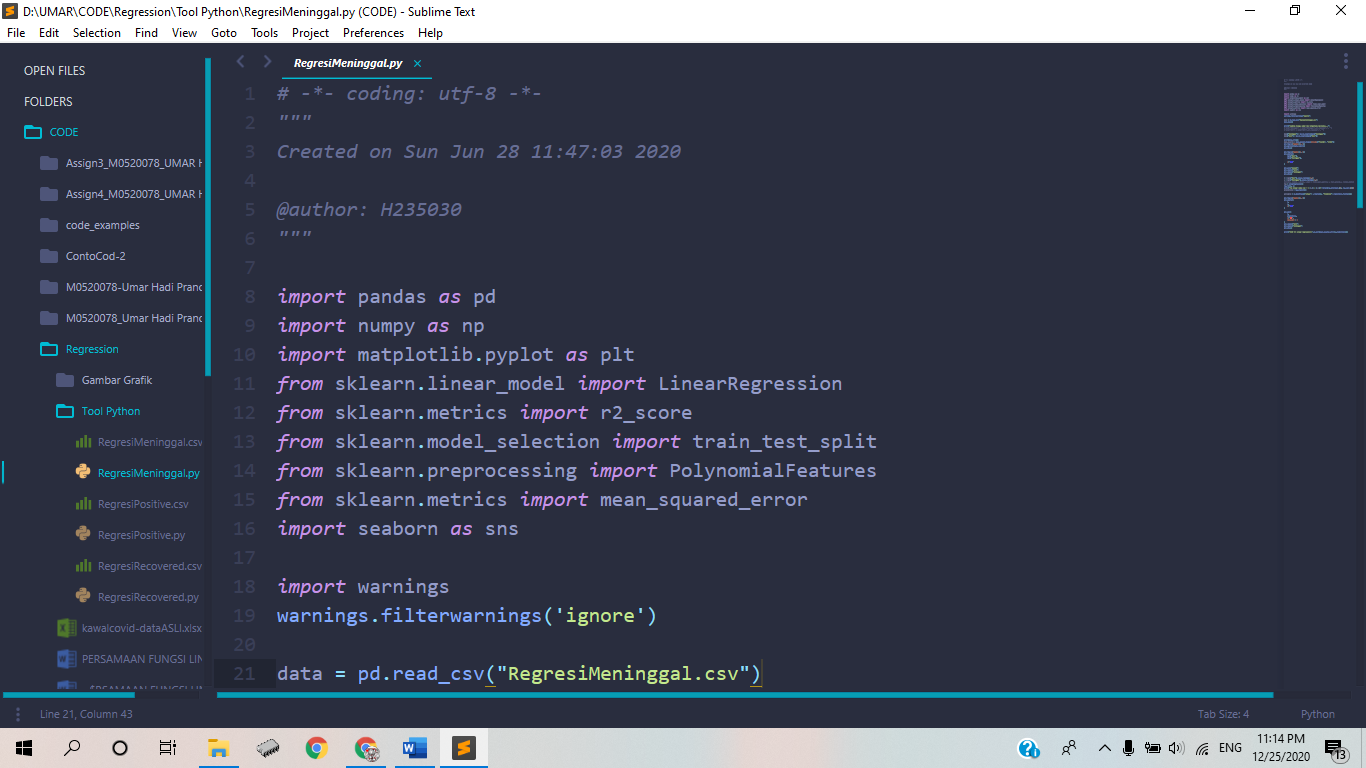


Dimana garis tersebut memiliki persamaan :

***Y = -269.37 + 5.9307X***

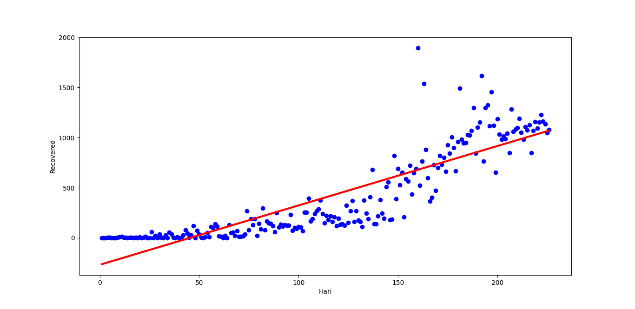
Pendekatan dilakukan dengan menggunakan suatu program yang dirancang atau disusun dengan menggunakan bahasa Python dengan versi 3.9.1 berikut tata cara bagaimana mendapatkan fungsi persamaan tersebut.

* 1. Program RegresionRecovered,py dijalankan dengan program CMD pada windows dimana sebelumnya telah di install program atau compiler Python dan telah melakukan instalasi seaborn, numpy, OpenCV, scikit-learn dan matplotlib di dalam cmd dengan perintah pip install.
  2. Selanjutanya program akan mengimpor dahulu dengan perintah *import pandas as pd, import numpy as np, import matplotlib, pyplot as plt*, dan *import seaborn as sns*. Selanjutnya dari file-file sklearn(scikit-learn) diimpor beberapa fungsi agar dapat menjalankan suatu perintah untuk mengolah data dan menampilkan data dalam bentuk gambar dengan perintah *from sklearn.linear\_model import LinearRegression, from sklearn.metrics import r2\_score, from sklearn.model\_selection import train\_test\_split, from sklearn.preprocessing import PolynomialFeatures,* dan *from sklearn.metrics import mean\_squared\_error*



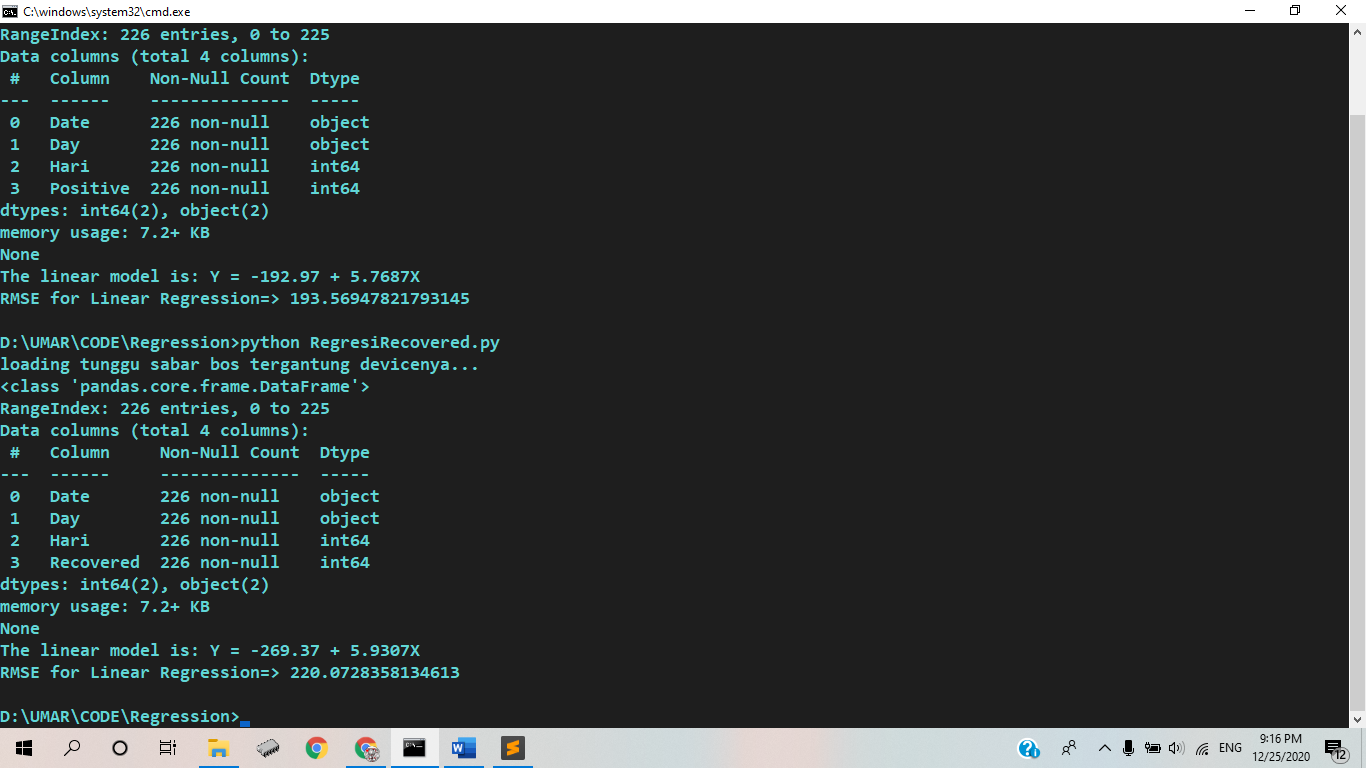
* 1. Selanjutnya akan dilakukan pembacaan data dari file “RegresiRecovered.csv” dimana file ini akan dipilah dari data yang berbentuk object(tabel yang berisi string atau huruf) atau int64(tabel yang berisi angka)
  2. Selanjutnya akan dilakukan pemilihan data yang akan digunakan pada grafik dengan tipe data angka dan tidak mengandung huruf yaitu data kolom “Hari” dan data kolom “Recovered” dari file .csv tersebut dengan perintah *data\_numeric = data.select\_dtypes(include=['float64', 'int64'])*, *data['Recovered'] = pd.to\_numeric(data['Recovered']),* dan *data['Hari'] = pd.to\_numeric(data['Hari'])*
  3. Selanjutnya akan dilakukan suatu penggambaran atau penyajian data berupa tabel grafik dengan titik X sebagai Hari dan titik Y sebagai Recovered dan akan dicetak dengan warna biru dengan perintah plt.scatter *(data['Hari'], data['Recovered'], c='blue')*
  4. Dan perhitungan regresi linear dilakukan dengan perintah *reg = LinearRegression() reg.fit(X, y)*. Dan ditampilkan pada program persamaan fungsi linear regresi dengan perintah *print("The linear model is: Y = {:.5} + {:.5}X".format(reg.intercept\_[0], reg.coef\_[0][0])) predictions = reg.predict(X)*
  5. Selanjutnya data disajikan dalam bentuk tabel grafik dengan menampilkan titik-titik biru(data Recovered) bewarna merah dan garis merah(garis regresi linear sesuai persamaan) dengan perintah *predictions = reg.predict(X) actvspred = d.DataFrame({'Actual': y.flatten(), 'Predicted': predictions.flatten()}) plt.figure(figsize=(4, 2)) plt.scatter(X,y,c='blue' ) plt.plot X, predictions, c='red', linewidth = 3) plt.xlabel("Hari") plt.ylabel("Recovered") plt.show().*

Sehingga ditampilkan seperti ini,

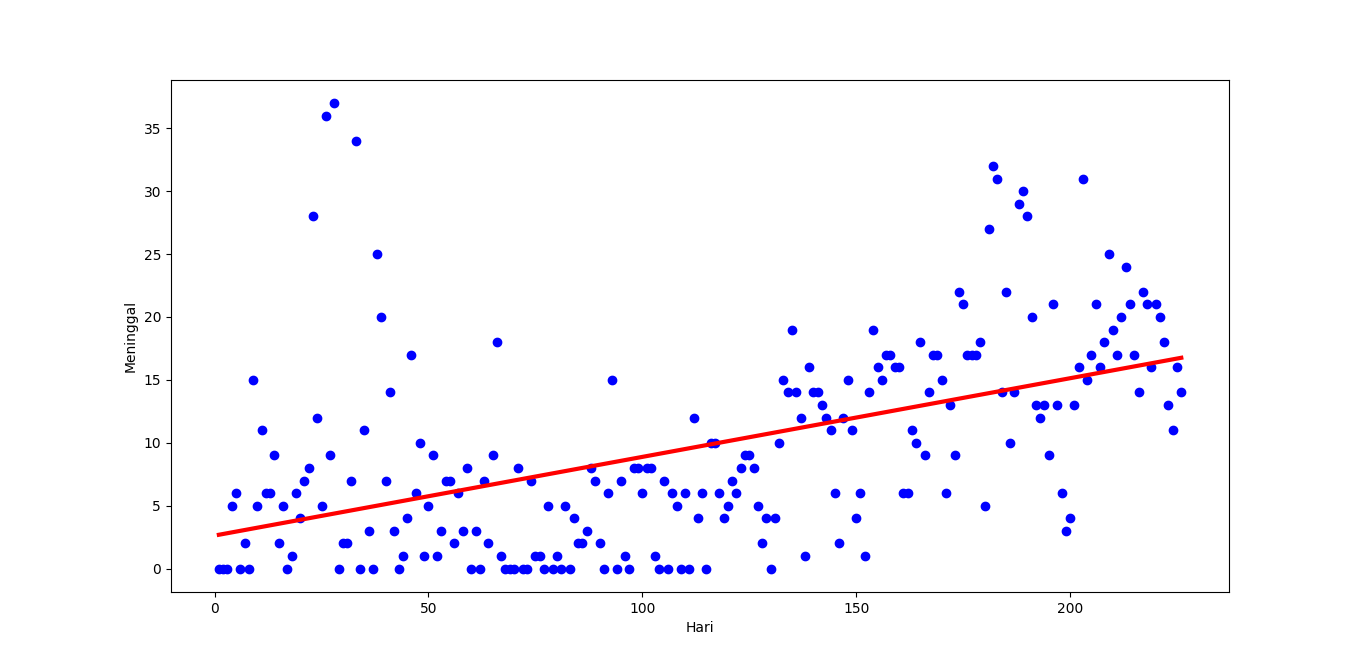


* 1. Di akhir ditampilkan Root Mean Squared Error (RMSE), yang digunakan sebagai indikator untuk mengukur kemiripan data dari titik-titik biru(data Recovered) bewarna merah dan garis merah(garis regresi linear sesuai persamaan dimana merupakan data juga) dengan perintah *print('RMSE for Linear Regression=>',np.sqrt(mean\_squared\_error(y,predictions))).*

Sehingga ditampilkan seperti ini,



1. **Mendapatkan Persamaan Fungsi Linear dari Data Meninggal**

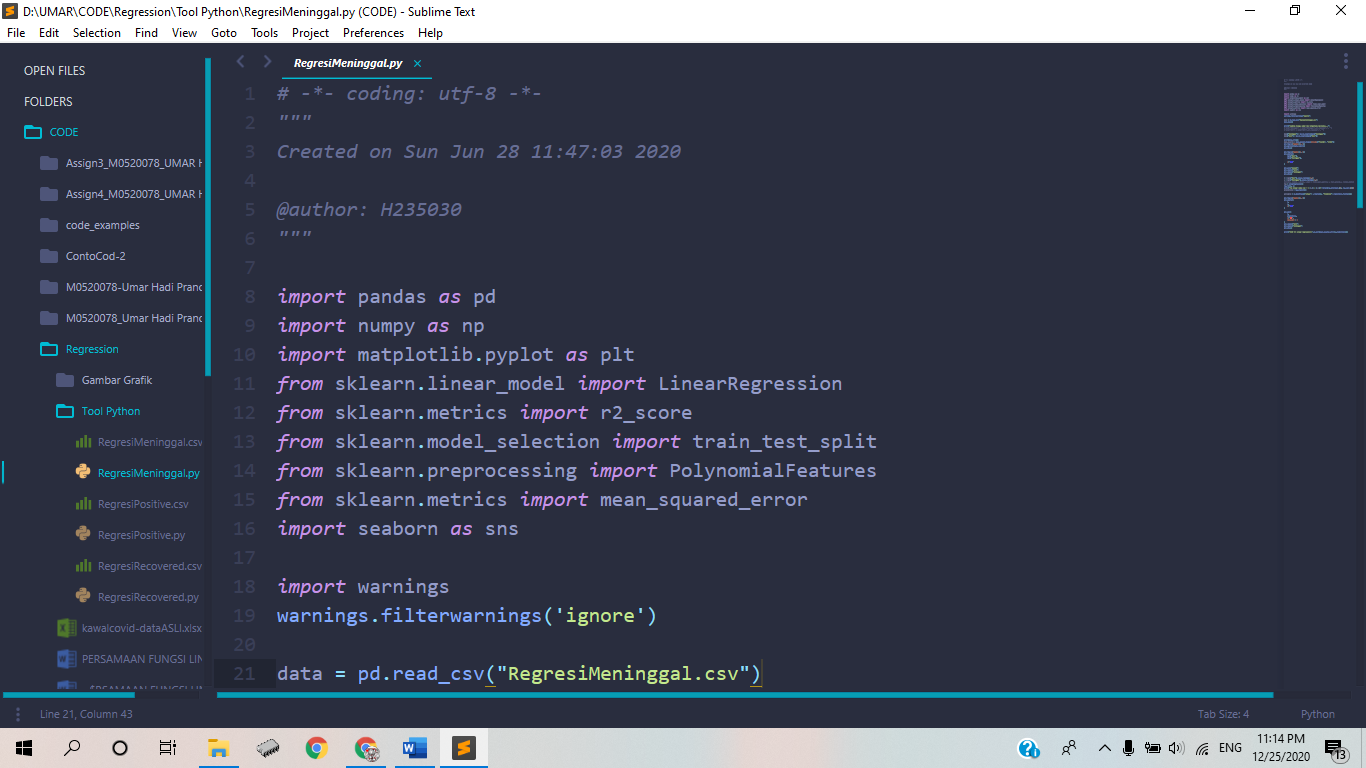


Dimana garis tersebut memiliki persamaan :

***Y = 2.6244 + 0.062527X***

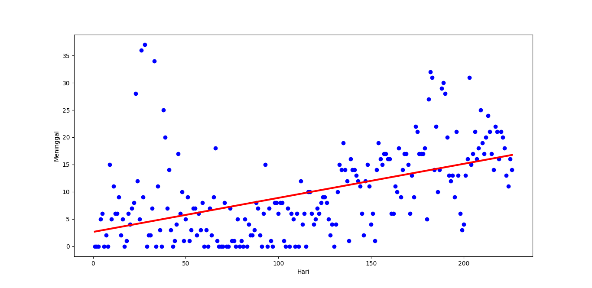
Pendekatan dilakukan dengan menggunakan suatu program yang dirancang atau disusun dengan menggunakan bahasa Python dengan versi 3.9.1 berikut tata cara bagaimana mendapatkan fungsi persamaan tersebut.

* 1. Program RegresionMeninggal,py dijalankan dengan program CMD pada windows dimana sebelumnya telah di install program atau compiler Python dan telah melakukan instalasi seaborn, numpy, OpenCV, scikit-learn dan matplotlib di dalam cmd dengan perintah pip install.
  2. Selanjutanya program akan mengimpor dahulu dengan perintah *import pandas as pd, import numpy as np, import matplotlib, pyplot as plt*, dan *import seaborn as sns*. Selanjutnya dari file-file sklearn(scikit-learn) diimpor beberapa fungsi agar dapat menjalankan suatu perintah untuk mengolah data dan menampilkan data dalam bentuk gambar dengan perintah *from sklearn.linear\_model import LinearRegression, from sklearn.metrics import r2\_score, from sklearn.model\_selection import train\_test\_split, from sklearn.preprocessing import PolynomialFeatures,* dan *from sklearn.metrics import mean\_squared\_error*



* 1. Selanjutnya akan dilakukan pembacaan data dari file “RegresiMeninggal.csv” dimana file ini akan dipilah dari data yang berbentuk object(tabel yang berisi string atau huruf) atau int64(tabel yang berisi angka)
  2. Selanjutnya akan dilakukan pemilihan data yang akan digunakan pada grafik dengan tipe data angka dan tidak mengandung huruf yaitu data kolom “Hari” dan data kolom “Meninggal” dari file .csv tersebut dengan perintah *data\_numeric = data.select\_dtypes(include=['float64', 'int64'])*, *data['Meninggal'] = pd.to\_numeric(data['Meninggal']),* dan *data['Hari'] = pd.to\_numeric(data['Hari'])*
  3. Selanjutnya akan dilakukan suatu penggambaran atau penyajian data berupa tabel grafik dengan titik X sebagai Hari dan titik Y sebagai Meninggal dan akan dicetak dengan warna biru dengan perintah plt.scatter *(data['Hari'], data['Meninggal'], c='blue')*
  4. Dan perhitungan regresi linear dilakukan dengan perintah *reg = LinearRegression() reg.fit(X, y)*. Dan ditampilkan pada program persamaan fungsi linear regresi dengan perintah *print("The linear model is: Y = {:.5} + {:.5}X".format(reg.intercept\_[0], reg.coef\_[0][0])) predictions = reg.predict(X)*
  5. Selanjutnya data disajikan dalam bentuk tabel grafik dengan menampilkan titik-titik biru(data Meninggal) bewarna merah dan garis merah(garis regresi linear sesuai persamaan) dengan perintah *predictions = reg.predict(X) actvspred = d.DataFrame({'Actual': y.flatten(), 'Predicted': predictions.flatten()}) plt.figure(figsize=(4, 2)) plt.scatter(X,y,c='blue' ) plt.plot X, predictions, c='red', linewidth = 3) plt.xlabel("Hari") plt.ylabel("Meninggal") plt.show().*

Sehingga ditampilkan seperti ini,



* 1. Di akhir ditampilkan Root Mean Squared Error (RMSE), yang digunakan sebagai indikator untuk mengukur kemiripan data dari titik-titik biru(data Meninggal) bewarna merah dan garis merah(garis regresi linear sesuai persamaan dimana merupakan data juga) dengan perintah *print('RMSE for Linear Regression=>',np.sqrt(mean\_squared\_error(y,predictions))).*

Sehingga ditampilkan seperti ini,

