**REFLEKSI MATAKULIAH PEMROGRAMAN DEKSTOP**

Disusun untuk Memenuhi Tugas Matakuliah Pemrograman Dekstop yang Dibimbing oleh Moh. Kautsar Sophan, S.Kom., M.MT.

****

Disusun oleh:

Achmad Farid Alfa Waid (190411100073)

**PROGRAM STUDI TEKNIK INFORMATIKA**

**FAKULTAS TEKNIK**

**UNIVERSITAS TRUNOJOYO MADURA**

**2020**

1. Buat tulisan yang menjelaskan hal-hal apa saja yang telah anda pelajari di kuliah ini!

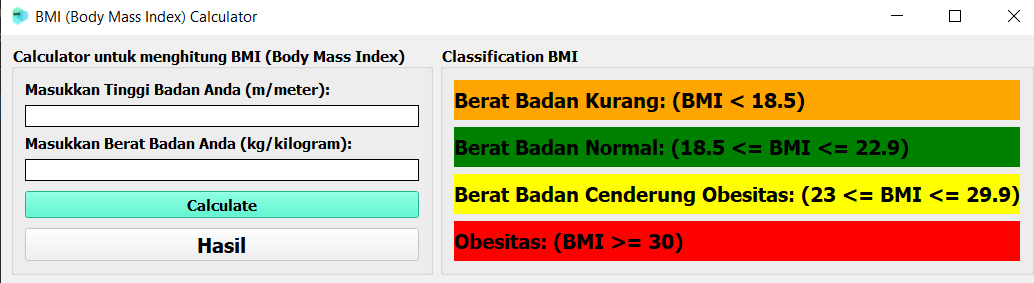
Pada matakuliah pemrograman desktop yang sudah berlangsung selama 5 pertemuan ini, sudah banyak sekali yang saya pelajari. Pertama pada awal pertemuan saya berkenalan dengan Pyqt5 yang ada di python dan bagaimana cara menjalankan pyqt untuk membuat sebuah tampilan GUI. Seiring berjalannya waktu saya belajar banyak syntax seperti syntak untuk membuat tampilan label, mengatur letak dan ukuran window, membuat button, dan mengatur button tersebut agar sesuai keinginan saya dengan syntax setstylesheet, membuat checkbox, membuat radio button, membuat line edit, membuat combo box, membuat group box, membuat spin box, membuat date time edit, membuat slider, membuat scroll bar, membuat dial, membuat signal, dan membuat berbagai macam layout. Pada waktu lain saya juga mencari referensi di internet untuk menggunakan syntax lain, seperti menconnectkan sebuah button ke fungsi lain ketika button tersebut di klik.

1. Buat 1 produk aplikasi bebas!

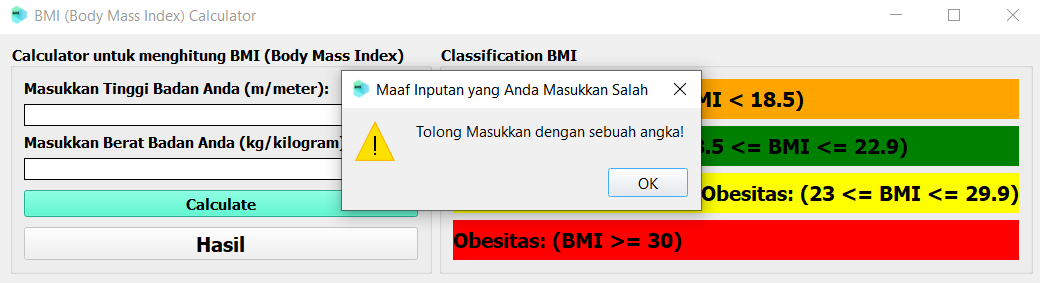
Pada tugas kali ini saya membuat sebuah aplikasi BMI (Body Mass Index) Calculator, jadi pada aplikasi ini user dapat menginputkan tinggi badan dan berat bandannya masing – masing yang nantinya akan dihitung oleh aplikasi ini dan akan di perlihatkan hasilnya.

1. Tuliskan listing program + jelaskan!
2. import sys
3. from PyQt5.QtWidgets import \*
4. from PyQt5.Qt import \*
5. #Membuat class yang didalamnya terdapat QWidget
6. class BMI(QWidget):
7. #Membuat fungsi init untuk inisialisasi class BMI
8. def \_\_init\_\_(self):
9. #untuk mengembalikan semua atribut dan method yang ada
10. super().\_\_init\_\_()
11. #memanggil fungsi BMI\_Calculate yang sudah dibuat agar ditampilkan hasilnya
12. self.BMI\_Calculate()
13. #membuat fungsi BMI calculate
14. def BMI\_Calculate(self):
15. #membuat grid layout dengan nama variabel grid
16. grid = QGridLayout()
17. #mengatur batas pada isi conten layout grid yaitu batas kana,kiri,atas,bawah bernilai 10
18. grid.setContentsMargins(10,10,10,10)
19. #membuat sebuag QGroupBox dengan nama variabel group1
20. group1 = QGroupBox("Calculator untuk menghitung BMI (Body Mass Index)",self)
21. #mengatur semua isi konten yang ada di group1 menjadi tulisan bold berwarna hitam dengan ukuran 15px
22. group1.setStyleSheet("font: bold 15px; color: black;")
23. #memasukkan group1 kedalam layout grid
24. grid.addWidget(group1, 1, 0)
25. #membuat QVBoxLayout dengan nama variabel vbox
26. vbox = QVBoxLayout()
27. #membuat vbox menjadi layout utama tetapi berada di dalam group1
28. group1.setLayout(vbox)
29. #membuat label1 dan akan dimasukkan ke dalam layout vbox
30. label1 = QLabel(self)
31. label1.setText("Masukkan Tinggi Badan Anda (m/meter):")
32. vbox.addWidget(label1)
34. #membuat QLineEdit dengan nama variabel height\_input
35. self.height\_input = QLineEdit(self)
36. #mengatur height\_input dengan border 0.5 px dan berwarna solid black
37. self.height\_input.setStyleSheet("border: 0.5px solid black;")
38. #memasukkan height\_input kedalam layout vbox
39. vbox.addWidget(self.height\_input)
40. #menambahkan addStretch ke dalam vbox agar tertata rapi
41. vbox.addStretch()
43. #membuat label1 dan akan dimasukkan ke dalam layout vbox
44. label2 = QLabel(self)
45. label2.setText("Masukkan Berat Badan Anda (kg/kilogram):")
46. vbox.addWidget(label2)
48. #membuat QLineEdit dengan nama variabel weight\_input
49. self.weight\_input = QLineEdit(self)
50. #mengatur weight\_input dengan border 0.5 px dan berwarna solid black
51. self.weight\_input.setStyleSheet("border: 0.5px solid black;")
52. #memasukkan weight\_input kedalam layout vbox
53. vbox.addWidget(self.weight\_input)
54. #menambahkan addStretch ke dalam vbox agar tertata rapi
55. vbox.addStretch(1)
56. #membuat sebuah button dengan nama variable cal\_button
57. self.cal\_button = QPushButton("Calculate", self)
58. #mengatur cal\_button dengan background berwarna #33f8c7
59. self.cal\_button.setStyleSheet("background-color: #33f8c7;")
60. #memasukkan cal\_button kedalam layout vbox
61. vbox.addWidget(self.cal\_button)
62. #menambahkan addStretch ke dalam vbox agar tertata rapi
63. vbox.addStretch(1)
64. #membuat sebuah button dengan nama variable hasil
65. self.hasil = QPushButton("Hasil", self)
66. ##mengatur hasil dengan font bold dan ukuran 20px dan berwarna hitam
67. self.hasil.setStyleSheet("font: bold 20px; color: black")
68. #mengatur button hasil dengan setDisabled agar tidak daat diklik
69. self.hasil.setDisabled(True)
70. #memasukkan button hasil ke dalam layout vbox
71. vbox.addWidget(self.hasil)
72. #membuat QGroupBox dengan nama variabel group2
73. group2 = QGroupBox("Classification BMI",self)
74. #mengatur group2 dengan font bol dan ukuran 15px dan berwarna hitam
75. group2.setStyleSheet("font: bold 15px; color: black;")
76. #memasukkan group2 kedalam layout utama grid
77. grid.addWidget(group2, 1, 1)
78. #membuat QVBoxLayout yang kedua untuk group2 dengan nama variabel vbox2
79. vbox2 = QVBoxLayout()
80. #mengatur vbox2 sebagai layout yang berada di dalam group2
81. group2.setLayout(vbox2)
82. #membuat label 3
83. label3 = QLabel(self)
84. label3.setText("Berat Badan Kurang: (BMI < 18.5)")
85. #mengatur label3 dengan background berwarna orange dan font bold dengan ukuran 20px dan warna tulisan hitam
86. label3.setStyleSheet("background-color: orange; font: bold 20px; color: black")
87. #memasukkan label3 kedalam layout vbox2
88. vbox2.addWidget(label3)
89. #membuat label 4
90. label4 = QLabel(self)
91. label4.setText("Berat Badan Normal: (18.5 <= BMI <= 22.9)")
92. #mengatur label3 dengan background berwarna hijau dan font bold dengan ukuran 20px dan warna tulisan hitam
93. label4.setStyleSheet("background-color: green; font: bold 20px; color: black")
94. #memasukkan label4 kedalam layout vbox2
95. vbox2.addWidget(label4)
96. #membuat label 5
97. label5 = QLabel(self)
98. label5.setText("Berat Badan Cenderung Obesitas: (23 <= BMI <= 29.9)")
99. #mengatur label3 dengan background berwarna kuning dan font bold dengan ukuran 20px dan warna tulisan hitam
100. label5.setStyleSheet("background-color: yellow; font: bold 20px; color: black")
101. #memasukkan label5 kedalam layout vbox2
102. vbox2.addWidget(label5)
103. #membuat label 6
104. label6 = QLabel(self)
105. label6.setText("Obesitas: (BMI >= 30)")
106. #mengatur label3 dengan background berwarna merah dan font bold dengan ukuran 20px dan warna tulisan hitam
107. label6.setStyleSheet("background-color: red; font: bold 20px; color: black")
108. #memasukkan label6 kedalam layout vbox2
109. vbox2.addWidget(label6)
110. #connect button to fungsi on\_click
111. self.cal\_button.clicked.connect(self.on\_click)
112. #membuat grid menjadi layout utama di window
113. self.setLayout(grid)
115. #membuat fungsi on\_click yang berisi sistem untuk menghitung nilai BMI
116. def on\_click(self):
117. #membuat QMessageBox dengan nama variabel notif
118. notif = QMessageBox()
119. try:
120. #membuat variabel weight dan height yang berisi inputan dari weight\_input dan height\_input yang akan dirubah menjadi sebuah float
121. weight = float(self.weight\_input.text())
122. height = float(self.height\_input.text())
123. #membuat variabel BMI untuk menghitung nilai BMI
124. BMI = weight/(height\*height)
125. #menampilkan nilai BMI pada button hasil
126. self.hasil.setText("Hasil Perhitungan BMI Anda: " + str("%.2f" %BMI))
127. ############ SISTEM BMI ############
128. #jika nilai BMI < 18.5
129. if BMI < 18.5:
130. #button hasil akan diubah menjadi background berwarna orange dan font bold berukuran wopx dan berwarna hitam
131. self.hasil.setStyleSheet("background-color: orange; font: bold 20px; color: black")
132. #menampilan sebuah meesage box dengan information
133. notif.information(self, "hasil", "Hasil Perhitungan BMI Anda: " + str("%.2f" %BMI) + "\nCategory: Berat Badan Anda Kurang", notif.Ok)
134. #jika nilai 18.5<=BMI<=22.9
135. elif 18.5<=BMI<=22.9:
136. #button hasil akan diubah menjadi background berwarna hijau dan font bold berukuran wopx dan berwarna hitam
137. self.hasil.setStyleSheet("background-color: green; font: bold 20px; color: black")
138. #menampilan sebuah meesage box dengan information
139. notif.information(self, "hasil", "Hasil Perhitungan BMI Anda: " + str("%.2f" %BMI) + "\nCategory: Berat Badan Anda Normal", notif.Ok)
140. #jika nilai 23<=BMI<=29.9
141. elif 23<=BMI<=29.9:
142. #button hasil akan diubah menjadi background berwarna kuning dan font bold berukuran wopx dan berwarna hitam
143. self.hasil.setStyleSheet("background-color: yellow; font: bold 20px; color: black")
144. #menampilan sebuah meesage box dengan information
145. notif.information(self, "hasil", "Hasil Perhitungan BMI Anda: " + str("%.2f" %BMI) + "\nCategory: Berat Badan Anda Cenderung Obesitas", notif.Ok)
146. #jika nilai 23<=BMI<=29.9
147. elif BMI>= 30:
148. #button hasil akan diubah menjadi background berwarna merah dan font bold berukuran wopx dan berwarna hitam
149. self.hasil.setStyleSheet("background-color: red; font: bold 20px; color: black")
150. #menampilan sebuah meesage box dengan information
151. notif.information(self, "hasil", "Hasil Perhitungan BMI Anda: " + str("%.2f" %BMI) + "\nCategory: Berat Badan Anda Obesitas", notif.Ok)
152. except:
153. #jika pada weight\_input dan height\_input diisi kosong
154. if str(self.weight\_input.text()) =="" and str(self.height\_input.text()) =="":
155. #maka meesage box akan menampilkan sebuah warning
156. notif.warning(self, "Maaf Inputan yang Anda Masukkan Salah", "Tolong Masukkan dengan sebuah angka! ", notif.Ok)
157. #jika pada weight\_input diisi kosong
158. elif str(self.weight\_input.text()) =="" and str(self.height\_input.text()) !="":
159. #maka meesage box akan menampilkan sebuah warning
160. notif.warning(self, "Masukkan Berat Badan Anda Dengan Benar!", "Tolong masukkan berat badan anda dengan sebuah angka! ", notif.Ok)
161. #jika pada height\_input diisi kosong
162. elif str(self.height\_input.text()) =="" and str(self.weight\_input.text()) !="":
163. #maka meesage box akan menampilkan sebuah warning
164. notif.warning(self, "Masukkan Tinggi Badan Anda Dengan Benar!", "Tolong masukkan tinggi badan anda dengan sebuah angka! ", notif.Ok)
165. #jika yang diiskan selain angka dan kosongana
166. else:
167. #maka meesage box akan menampilkan sebuah warning
168. notif.warning(self, "Maaf Inputan yang Anda Masukkan Salah", "Tolong Masukkan dengan sebuah angka! ", notif.Ok)

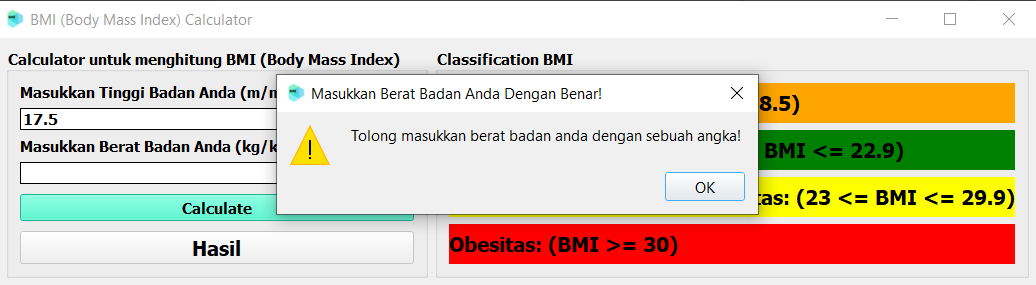
171. if \_\_name\_\_ == '\_\_main\_\_':
172. #Inisisalisai pyqt
173. app = QApplication(sys.argv)
174. #mengatur style di window menjadi style fusion
175. app.setStyle("fusion")
176. #menambahkan icon pada window
177. app.setWindowIcon(QIcon('bmi\_logo.ico'))
178. #membuat variabel ex yang berisi class BMI
179. ex = BMI()
180. #Menentukan ukuran window dan title untuk menampilkan
181. ex.setGeometry(100,100,500,250)
182. #membuat judul window
183. ex.setWindowTitle("BMI (Body Mass Index) Calculator")
184. #menampilan isi dari variabel ex
185. ex.show()
186. #membuat system exit
187. sys.exit(app.exec\_())
188. Capture hasil running!



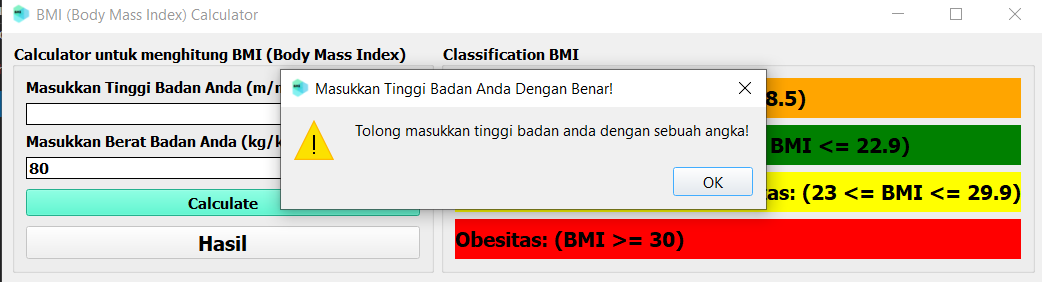
Pada saat tinggi badan dan berat badan bersisi kosongan.



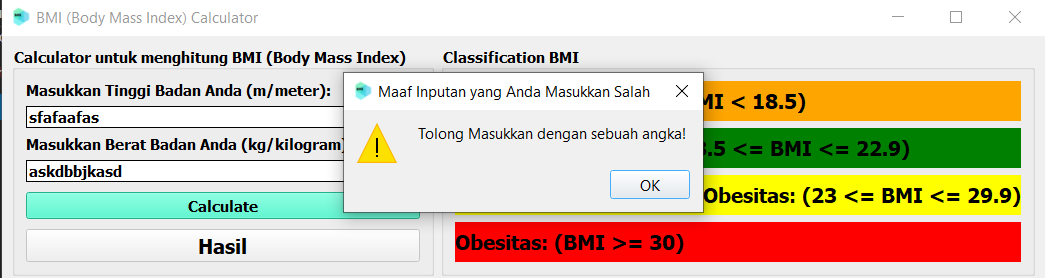
Pada saat tinggi badan diisi dengan angka tetapi berat badan kosongan.



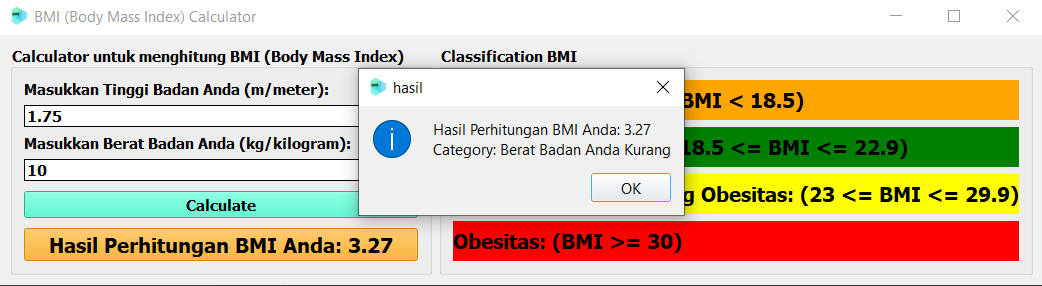
Pada saat berat badan diisi dengan angka dan tinggi badan kosongan.



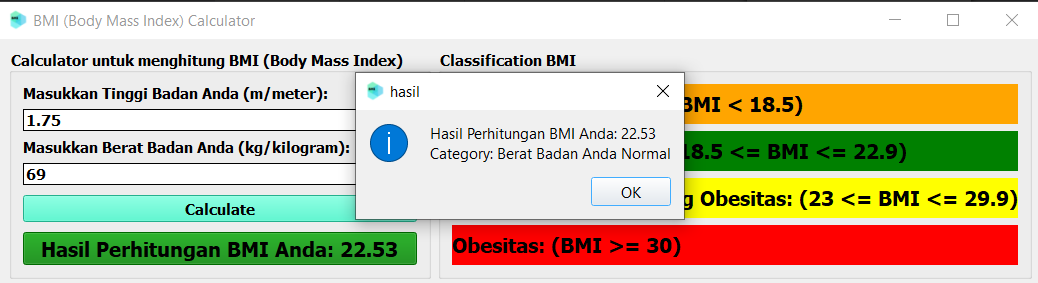
Pada saat tinggi badan dan berat badan diisikan dengan selain angka.



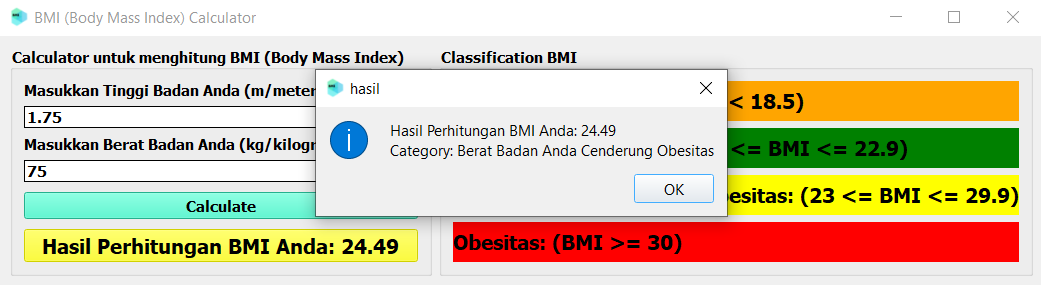
Pada saat nilai BMI < 18.5, button hasil berubah menjadi warna orange dan akan memunculkan nilai BMI user, dan akan ada notif seperti berikut.



Pada saat nilai 18.5 <= BMI <= 22.9, button hasil berubah menjadi warna hijau dan akan memunculkan nilai BMI user, dan akan ada notif seperti berikut.



Pada saat nilai 23 <= BMI <= 29.9, button hasil berubah menjadi warna kuning dan akan memunculkan nilai BMI user, dan akan ada notif seperti berikut.



Pada saat nilai BMI >= 30, button hasil berubah menjadi warna merah dan akan memunculkan nilai BMI user, dan akan ada notif seperti berikut.

