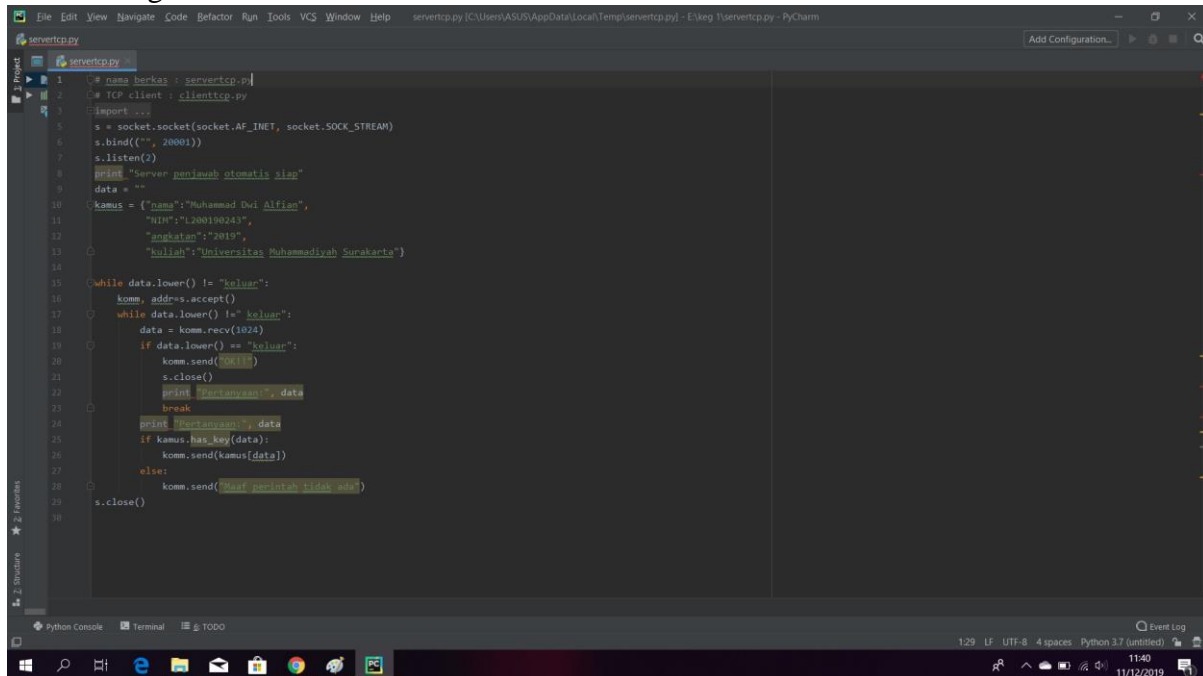


PRAKTIKUM 10

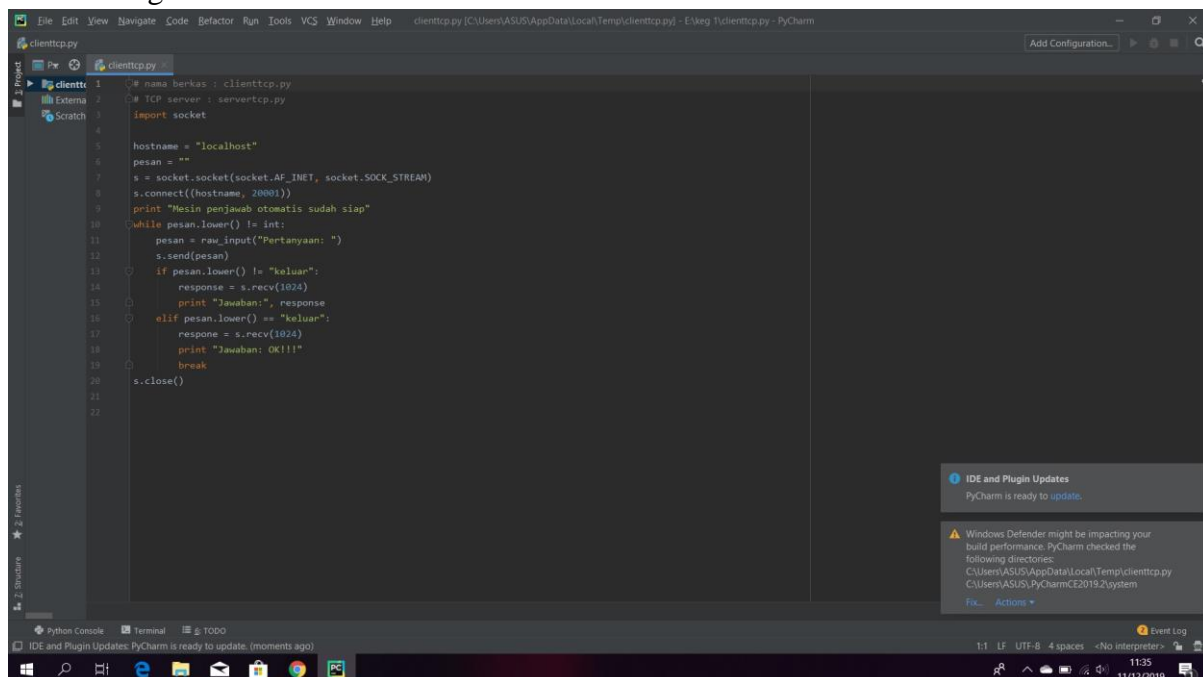
Kegiatan 1

Program server



```
1 # nama berkas : servertcp.py
2 # TCP client : clienttcp.py
3 import socket
4
5 s = socket.socket(socket.AF_INET, socket.SOCK_STREAM)
6 s.bind(('', 20001))
7 s.listen(2)
8 print "Server penjawab otomatis siap"
9 data = ""
10
11 kamus = {'name': "Muhammad Dwi Alfian",
12         "NIM": "12001902431",
13         "angkatan": "2019",
14         "kuliah": "Universitas Muhammadiyah Surakarta"}
15
16 while data.lower() != "keluar":
17     koms, addrs = s.accept()
18     while data.lower() != "keluar":
19         data = koms.recv(1024)
20         if data.lower() == "keluar":
21             koms.send("OK!!!")
22             s.close()
23             print "Pertanyaan:", data
24             break
25         print "Pertanyaan:", data
26         if kamus.has_key(data):
27             koms.send(kamus[data])
28         else:
29             koms.send("Maaf perintah tidak ada")
30     s.close()
```

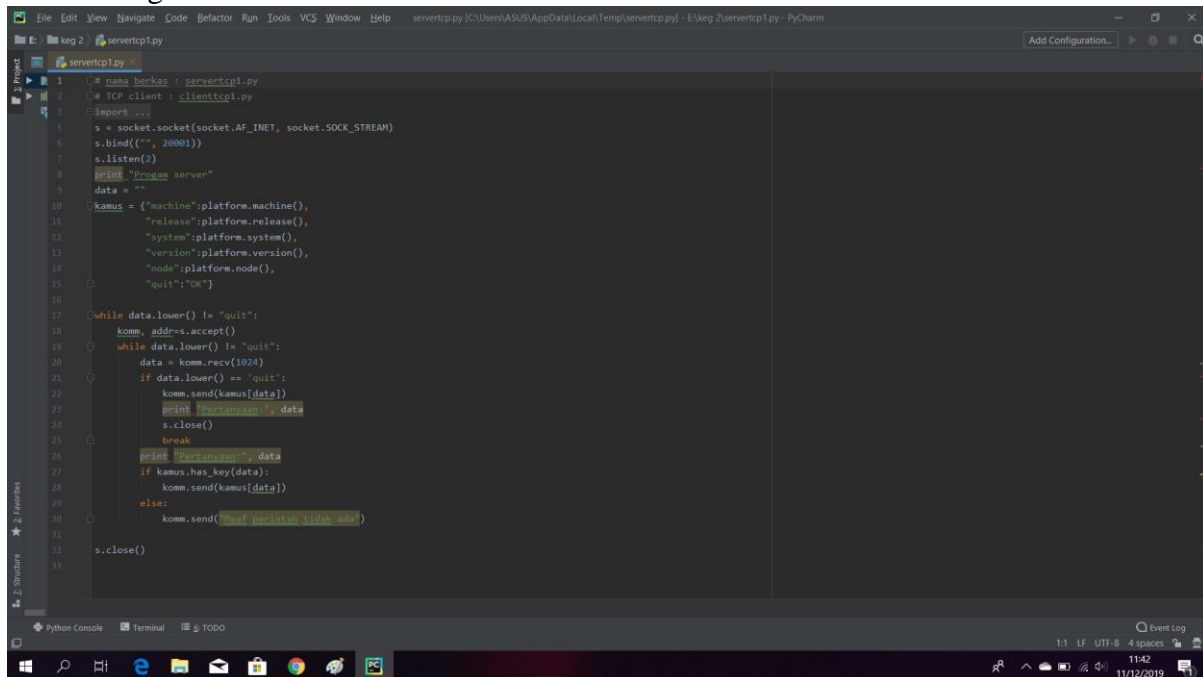
Program Client



```
1 # nama berkas : clienttcp.py
2 # TCP server : servertcp.py
3 import socket
4
5 hostname = "localhost"
6 pesan = ""
7
8 s = socket.socket(socket.AF_INET, socket.SOCK_STREAM)
9 s.connect((hostname, 20001))
10 print "Mesin penjawab otomatis sudah siap"
11
12 while pesan.lower() != "int":
13     pesan = raw_input("Pertanyaan: ")
14     s.send(pesan)
15     if pesan.lower() != "keluar":
16         response = s.recv(1024)
17         print "Jawaban:", response
18     elif pesan.lower() == "keluar":
19         response = s.recv(1024)
20         print "Jawaban: OK!!!"
21         break
22     s.close()
```

Kegiatan 2

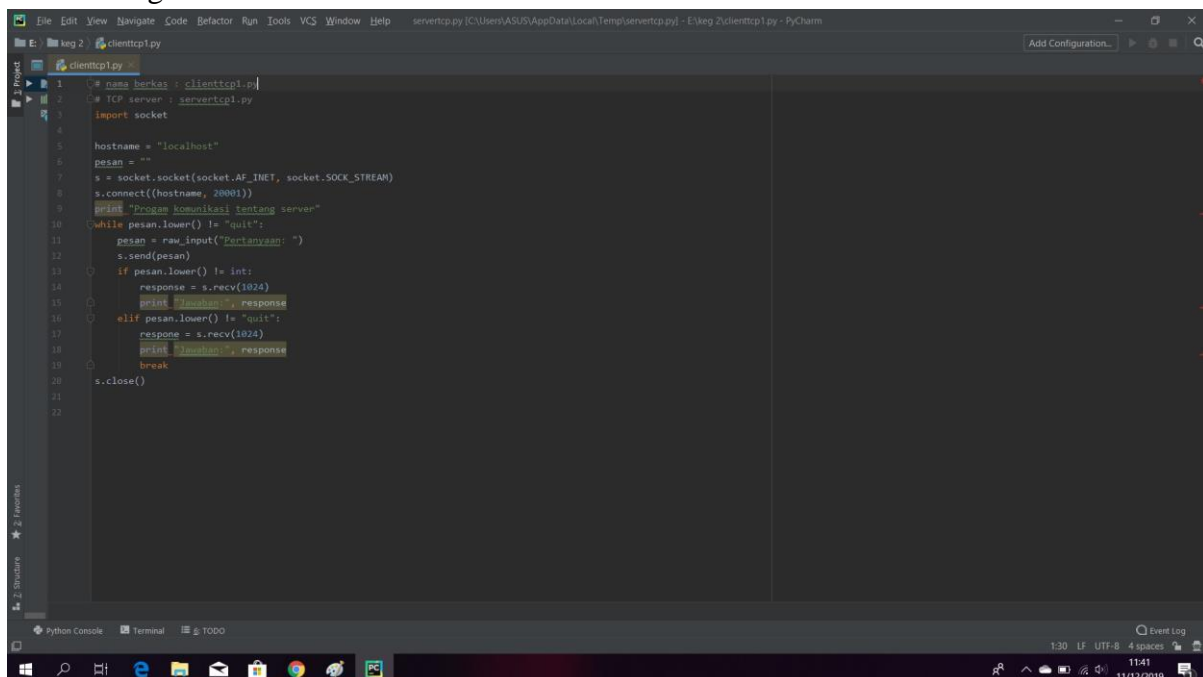
Program Server



The screenshot shows the PyCharm IDE with a file named `servertcp1.py` open. The code is a Python script for a TCP server. It imports the `socket` module and defines a `main` function. The server binds to `localhost` on port `20001` and listens for incoming connections. When a connection is accepted, it prints a message and receives data from the client. It then checks if the data is `"quit"`. If not, it sends a response containing system information (machine, release, system, version, node) and the received data. If the data is `"quit"`, it sends a response and closes the connection. The server also has a `while` loop to handle multiple connections.

```
1 # nama berkas : servertcp1.py
2 # TCP client : clienttcp1.py
3 import socket
4
5 s = socket.socket(socket.AF_INET, socket.SOCK_STREAM)
6 s.bind(("*", 20001))
7 s.listen(2)
8 print "Program server"
9 data = ""
10
11 kamus = {"machine":platform.machine(),
12         "release":platform.release(),
13         "system":platform.system(),
14         "version":platform.version(),
15         "node":platform.node(),
16         "quit":"OK"}
17
18 while data.lower() != "quit":
19     komm, addr=s.accept()
20     while data.lower() != "quit":
21         data = komm.recv(1024)
22         if data.lower() == "quit":
23             komm.send(kamus[data])
24             print "Pertanyaan:", data
25             s.close()
26             break
27         print "Pertanyaan:", data
28         if kamus.has_key(data):
29             komm.send(kamus[data])
30         else:
31             komm.send("Maaf pertanyaan tidak ada")
32     s.close()
33
```

Program Client

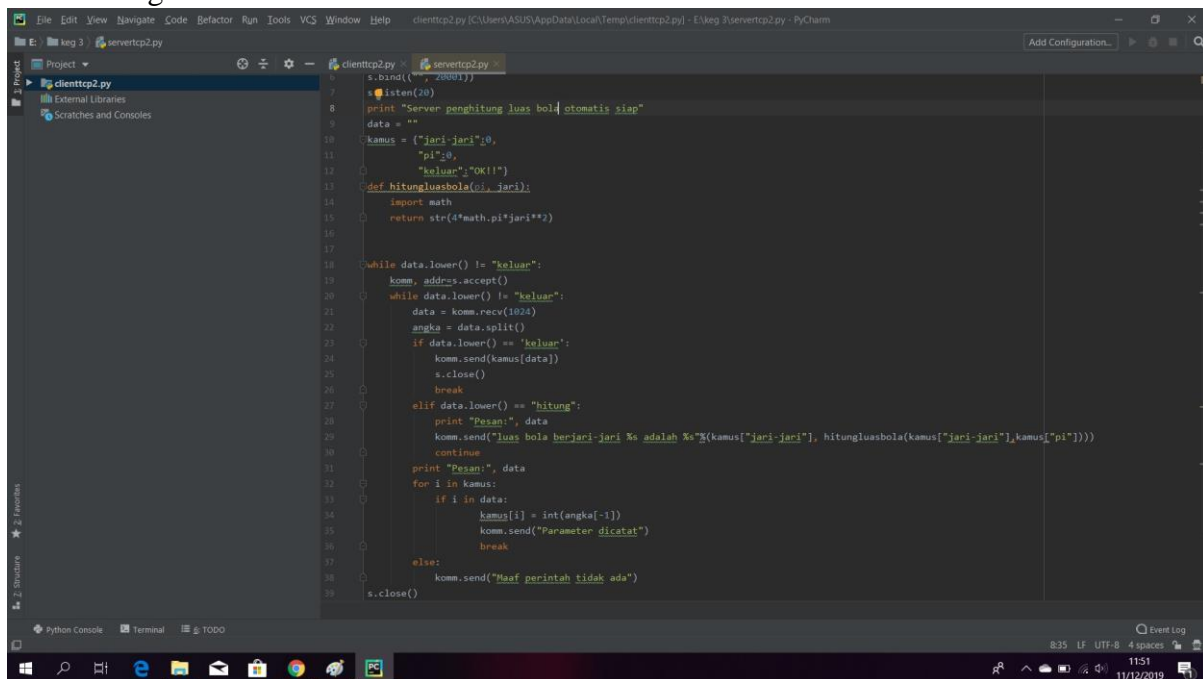


The screenshot shows the PyCharm IDE with a file named `clienttcp1.py` open. The code is a Python script for a TCP client. It imports the `socket` module and defines a `main` function. The client connects to `localhost` on port `20001`. It then enters a `while` loop where it prompts the user for a question. If the user enters `"quit"`, it sends the message and closes the connection. Otherwise, it sends the question and receives a response. It then prints the response and checks if it is `"quit"`. If not, it prints the response and continues the loop. If the response is `"quit"`, it prints the response and closes the connection.

```
1 # nama berkas : clienttcp1.py
2 # TCP server : servertcp1.py
3 import socket
4
5 hostname = "localhost"
6 pesan = ""
7 s = socket.socket(socket.AF_INET, socket.SOCK_STREAM)
8 s.connect((hostname, 20001))
9 print "Program komunikasi tentang server"
10
11 while pesan.lower() != "quit":
12     pesan = raw_input("Pertanyaan: ")
13     s.send(pesan)
14     if pesan.lower() != "quit":
15         response = s.recv(1024)
16         print "Jawaban:", response
17     elif pesan.lower() != "quit":
18         response = s.recv(1024)
19         print "Jawaban:", response
20         break
21 s.close()
22
```

Kegiatan 3

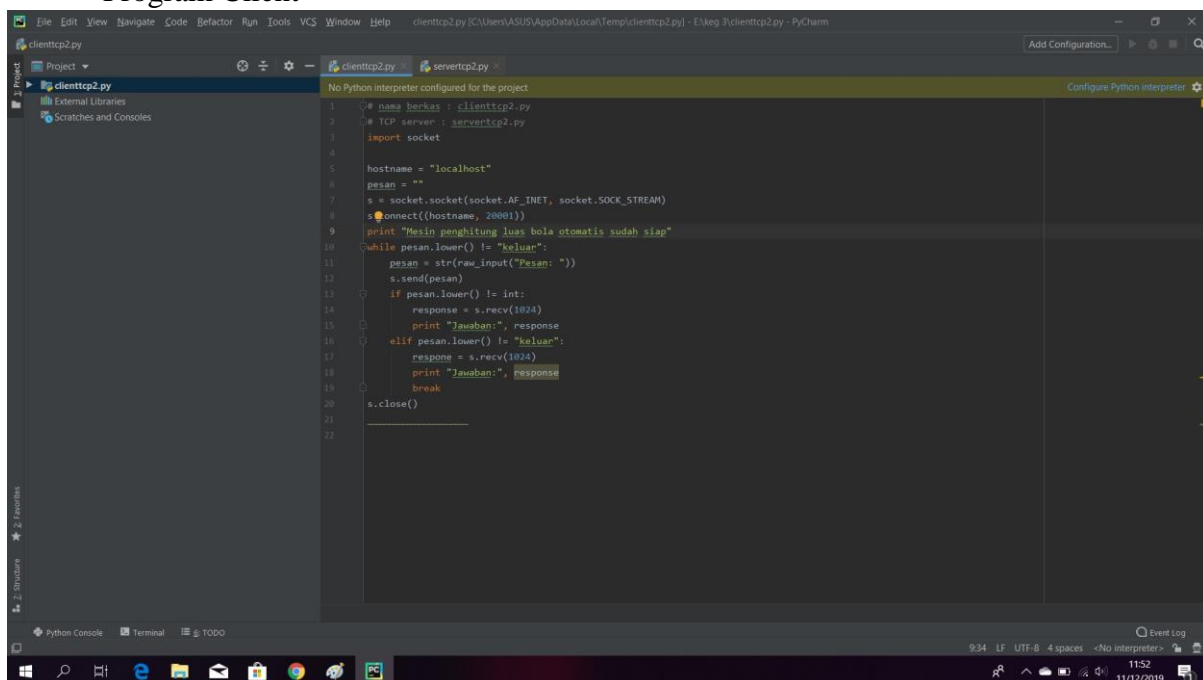
Program Server



The screenshot shows the PyCharm IDE with the 'clienttcp2.py' file open. The code is a Python script for a server that listens on port 20001. It prints a message 'Server penghitung luas bola otomatis siap'. It then enters a loop where it accepts connections. If the received data is 'keluar', it sends back 'OK!!'. If the data is 'hitung', it calculates the area of a circle using the formula $4 * \pi * r^2$ and sends the result. If the data is 'Pesan:', it sends back the message. If the data is 'Parameter dicatat', it sends back 'Parameter dicatat'. If the data is 'Maaf perintah tidak ada', it sends back 'Maaf perintah tidak ada'. The script ends with 's.close()'.

```
1 s.bind(('', 20001))
2 s.listen(20)
3 print "Server penghitung luas bola otomatis siap"
4 data = ""
5
6 kamus = {"jari-jari":0,
7         "pi":0,
8         "keluar": "OK!!"}
9
10 def hitungluasbola(r, pi):
11     import math
12     return str(4*math.pi*r**2)
13
14 while data.lower() != "keluar":
15     k, addr = s.accept()
16     while data.lower() != "keluar":
17         data = k.recv(1024)
18         angka = data.split()
19         if data.lower() == "keluar":
20             k.send(kamus[data])
21             k.close()
22             break
23         elif data.lower() == "hitung":
24             print "Pesan:", data
25             k.send("luas bola berjari-jari %s adalah %s" % (kamus["jari-jari"], hitungluasbola(kamus["jari-jari"], kamus["pi"])))
26             continue
27             print "Pesan:", data
28             for i in kamus:
29                 if i in data:
30                     kamus[i] = int(angka[-1])
31                     k.send("Parameter dicatat")
32                     break
33             else:
34                 k.send("Maaf perintah tidak ada")
35     s.close()
```

Program Client



The screenshot shows the PyCharm IDE with the 'clienttcp2.py' file open. The code is a Python script for a client that connects to the server on port 20001. It prints a message 'Mesin penghitung luas bola otomatis sudah siap'. It then enters a loop where it sends data to the server. If the received data is 'keluar', it sends back 'OK!!'. If the data is 'hitung', it calculates the area of a circle using the formula $4 * \pi * r^2$ and sends the result. If the data is 'Pesan:', it sends back the message. If the data is 'Parameter dicatat', it sends back 'Parameter dicatat'. If the data is 'Maaf perintah tidak ada', it sends back 'Maaf perintah tidak ada'. The script ends with 's.close()'.

```
1 # nama berkas : clienttcp2.py
2 # TCP server : servertcp2.py
3 import socket
4
5 hostname = "localhost"
6 pesan = ""
7 s = socket.socket(socket.AF_INET, socket.SOCK_STREAM)
8 s.connect((hostname, 20001))
9 print "Mesin penghitung luas bola otomatis sudah siap"
10 while pesan.lower() != "keluar":
11     pesan = str(raw_input("Pesan: "))
12     s.send(pesan)
13     if pesan.lower() != "keluar":
14         response = s.recv(1024)
15         print "Jawaban:", response
16     elif pesan.lower() != "hitung":
17         response = s.recv(1024)
18         print "Jawaban:", response
19     break
20 s.close()
```

