

Nama : Sarah Ayu Rahmawati
Nim : 20210040084
Prodi : Teknik Informatika
Kelas : TI21A

KECERDASAN BUATAN TUGAS SESI 12

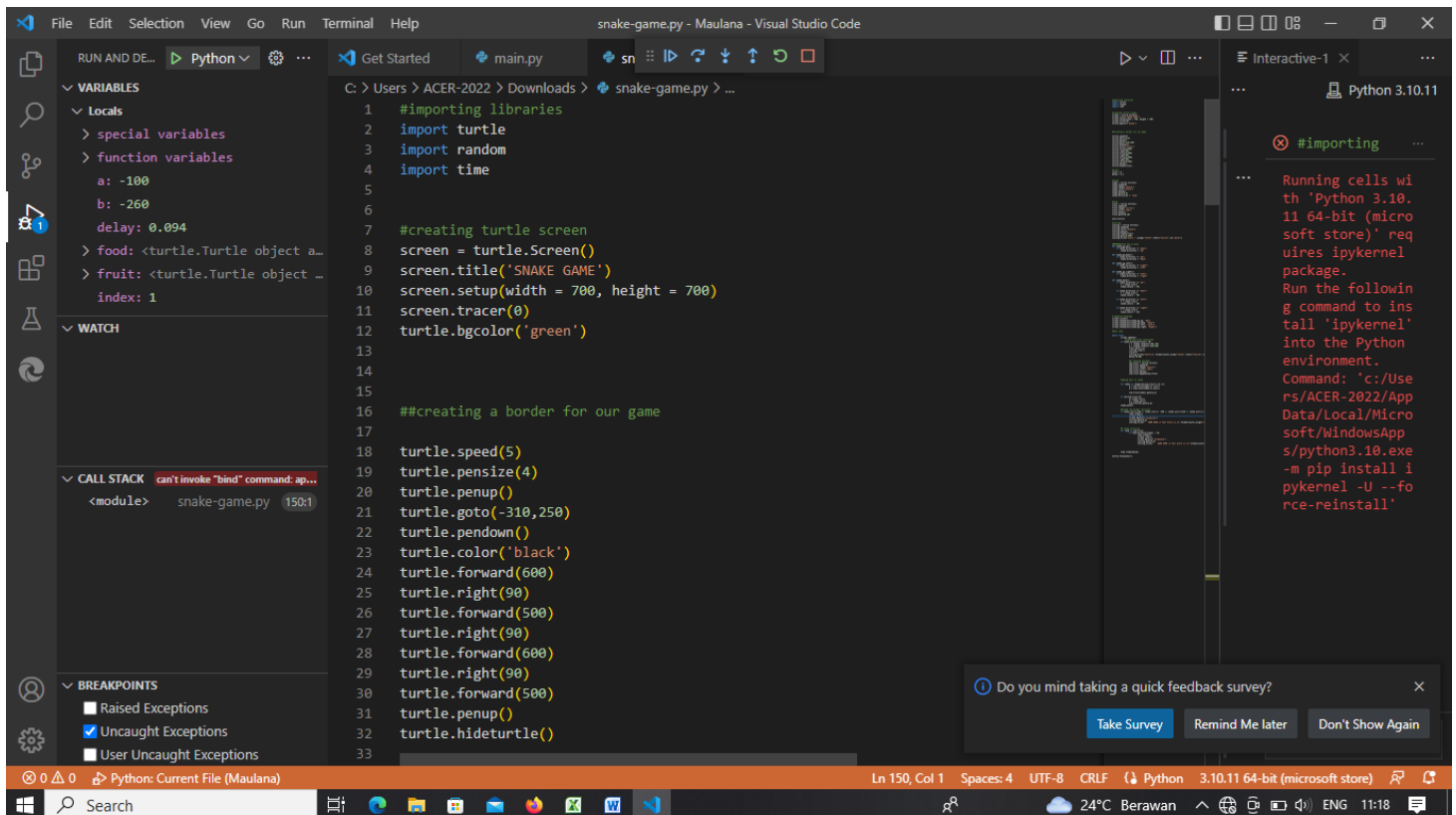
Pilihlah salah satu Perancangan game yang anda sukai, screenshot setiap tahapan kerja dan hasilnya

Jawab :

Disini saya memilih Snake game. Snake adalah nama umum untuk konsep permainan video dimana pemain mengendalikan sebuah garis yang tumbuh memanjang, dengan garis itu sendiri menjadi rintangan utama. Konsep tersebut bermula pada permainan arkade 1976 Blockade, dan kemudian membuat Snake menjadi memiliki ratusan versi (beberapa diantaranya memakai kata snake (ular) atau worm (ulat) pada judulnya) untuk beberapa wadah.

Si pemain akan mengendalikan sebuah makhluk yang menyerupai ular yang akan bergerak mengitari sebuah bidang berbentuk kotak, dengan tujuan mengambil makanan yang aslinya berbentuk dot atau titik. Selama bermain, si pemain harus berusaha untuk tidak menabrak dinding atau ekornya sendiri dan itu akan semakin susah, karena setiap kali si pemain memakan makanan, ekornya akan bertambah panjang. Kontrol-pun sangat mudah, yakni hanya atas, bawah, kiri dan kanan, ular akan berjalan secara otomatis dan tidak dapat dihentikan.

Adapun source code nya adalah sebagai berikut



```
1 #importing libraries
2 import turtle
3 import random
4 import time
5
6 #creating turtle screen
7 screen = turtle.Screen()
8 screen.title('SNAKE GAME')
9 screen.setup(width = 700, height = 700)
10 screen.tracer(0)
11 turtle.bgcolor('green')
12
13
14
15
16 ##creating a border for our game
17
18 turtle.speed(5)
19 turtle.pensize(4)
20 turtle.penup()
21 turtle.goto(-310,250)
22 turtle.pendown()
23 turtle.color('black')
24 turtle.forward(600)
25 turtle.right(90)
26 turtle.forward(500)
27 turtle.right(90)
28 turtle.forward(600)
29 turtle.right(90)
30 turtle.forward(500)
31 turtle.penup()
32 turtle.hideturtle()
33
```

Visual Studio Code interface showing a Python file named `snake-game.py` in the editor. The file is located at `C:\Users\ACER-2022\Downloads\snake-game.py`. The editor displays the following code:

```
34 #score
35 score = 0
36 delay = 0.1
37
38 #snake
39 snake = turtle.Turtle()
40 snake.speed(0)
41 snake.shape('square')
42 snake.color("black")
43 snake.penup()
44 snake.goto(0,0)
45 snake.direction = 'stop'
46
47 #food
48
49 fruit = turtle.Turtle()
50 fruit.speed(0)
51 fruit.shape('circle')
52 fruit.color('red')
53 fruit.penup()
54 fruit.goto(30,30)
55
56 old_fruit=[]
57
58 #scoring
59 scoring = turtle.Turtle()
60 scoring.speed(0)
61 scoring.color("black")
62 scoring.penup()
63 scoring.hideturtle()
64 scoring.goto(0,300)
65 scoring.write("Score :",align="center",font=("Courier",24,"bold"))
66
```

The left sidebar shows the **VARIABLES** pane with the following content:

- Locals**
 - special variables
 - function variables
 - a: -100
 - b: -260
 - delay: 0.094
 - food: <turtle.Turtle object a...
 - fruit: <turtle.Turtle object ...
index: 1
- WATCH**
- CALL STACK** (can't invoke "bind" command: ap...)
 - <module> snake-game.py (150:1)
- BREAKPOINTS**
 - ☐ Raised Exceptions
 - ☒ Uncaught Exceptions
 - ☐ User Uncaught Exceptions

The right sidebar shows the **Interactive-1** pane with the following content:

```
#importing ...
...
Running cells with 'Python 3.10.11 64-bit (microsoft store)' requires ipykernel package.
Run the following command to install 'ipykernel' into the Python environment.
Command: 'c:/Users/ACER-2022/AppData/Local/Microsoft/WindowsApps/python3.10.exe -m pip install ipykernel -U --force-reinstall'
```

A feedback survey dialog box is displayed in the bottom right corner with the text: "Do you mind taking a quick feedback survey?". The dialog has three buttons: "Take Survey", "Remind Me later", and "Don't Show Again".

Visual Studio Code interface showing a Python file named `snake-game.py` in the editor. The file is located at `C:\Users\ACER-2022\Downloads\snake-game.py`. The editor displays the following code:

```
69 #####define how to move
70 def snake_go_up():
71     if snake.direction != "down":
72         snake.direction = "up"
73
74 def snake_go_down():
75     if snake.direction != "up":
76         snake.direction = "down"
77
78 def snake_go_left():
79     if snake.direction != "right":
80         snake.direction = "left"
81
82 def snake_go_right():
83     if snake.direction != "left":
84         snake.direction = "right"
85
86 def snake_move():
87     if snake.direction == "up":
88         y = snake.ycor()
89         snake.sety(y + 20)
90
91     if snake.direction == "down":
92         y = snake.ycor()
93         snake.sety(y - 20)
94
95     if snake.direction == "left":
96         x = snake.xcor()
97         snake.setx(x - 20)
98
99     if snake.direction == "right":
100         x = snake.xcor()
101         snake.setx(x + 20)
```

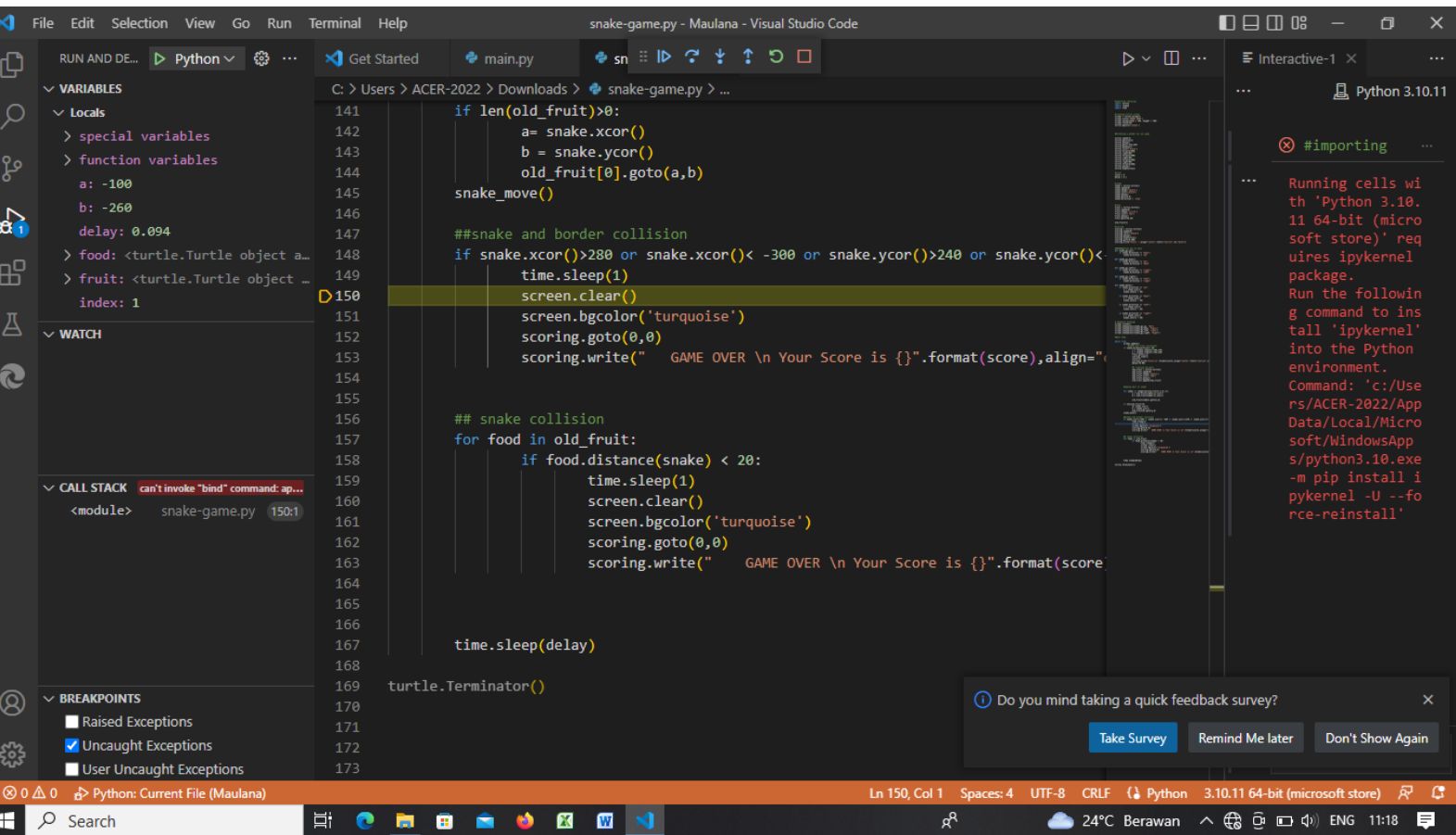
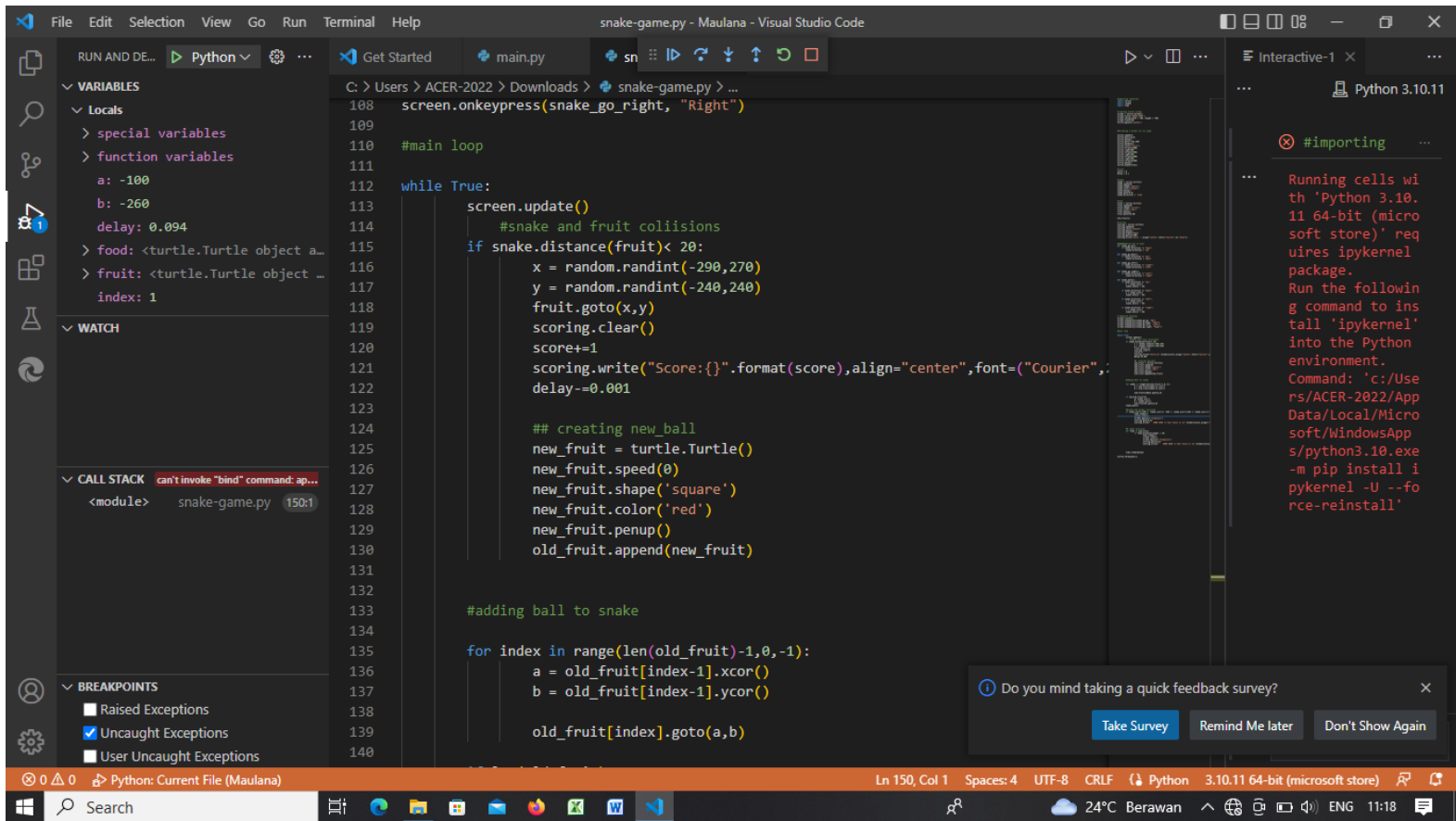
The left sidebar shows the **VARIABLES** pane with the following content:

- Locals**
 - special variables
 - function variables
 - a: -100
 - b: -260
 - delay: 0.094
 - food: <turtle.Turtle object a...
 - fruit: <turtle.Turtle object ...
index: 1
- WATCH**
- CALL STACK** (can't invoke "bind" command: ap...)
 - <module> snake-game.py (150:1)
- BREAKPOINTS**
 - ☐ Raised Exceptions
 - ☒ Uncaught Exceptions
 - ☐ User Uncaught Exceptions

The right sidebar shows the **Interactive-1** pane with the following content:

```
#importing ...
...
Running cells with 'Python 3.10.11 64-bit (microsoft store)' requires ipykernel package.
Run the following command to install 'ipykernel' into the Python environment.
Command: 'c:/Users/ACER-2022/AppData/Local/Microsoft/WindowsApps/python3.10.exe -m pip install ipykernel -U --force-reinstall'
```

A feedback survey dialog box is displayed in the bottom right corner with the text: "Do you mind taking a quick feedback survey?". The dialog has three buttons: "Take Survey", "Remind Me later", and "Don't Show Again".



Berikut adalah tahapan kerja dalam membuat game Snake:

- Menyiapkan lingkungan kerja:
Menginstall Python dan library Pygame.
Membuat file Python baru untuk mengkodekan game Snake.
- Menginisialisasi Pygame:
Mengimpor library Pygame.
Menginisialisasi Pygame dan membuat game layar.
- merekam variabel dan konstanta:
Menentukan ukuran layar.
Menentukan ukuran blok untuk ular dan makanan.
Menentukan kecepatan pergerakan ular.
- Membuat kelas Snake:
Membuat konstruktor untuk mengatur posisi awal ular dan panjangnya.
Membuat metode menggerakkan ular dan mengubah arahnya.
Membuat metode untuk memeriksa tabrakan ular dengan dinding atau tubuhnya sendiri.
- Membuat kelas Apple:
Membuat konstruktor untuk mengatur posisi awal apel secara acak.
Membuat metode untuk memeriksa apakah ular memakan apel.
- langkah-langkah masukan pengguna:
Gunakan event loop untuk menangkap pengguna masukan (misalnya, tombol panah) dan menggerakkan ular.
- Menggambar elemen-elemen permainan:
Gunakan metode `pygame.draw.rect()` untuk menggambar ular dan apel.
Gunakan metode `pygame.display.flip()` untuk memperbarui tampilan game setiap frame.
- permainan berjalan:
Gunakan loop utama untuk menjalankan game.

Memeriksa kondisi game over (misalnya, saat ular menabrak dinding atau tubuhnya sendiri)

