Jpeg ConnectFourIcon

Gif ConnectFourEmptySpot

Gif ConnectFourRedToken

Gif ConnectFourBlueToken

Gif ConnectFourGreenToken

Gif ConnectFourYellowToken

File ConnectFour

import ConnectFourView from ConnectFourStrategy

class Controller()

* **def \_\_init\_\_(self, view):**
  + *self.\_\_strategy\_pattern = ConnectFourView()*
  + *self.\_\_view = \_\_strategy\_pattern.view()*
  + *self.\_\_players\_range = [2][5]*
  + *self.\_\_player\_amount = 2*
  + *self.\_\_player\_list = []*
* **def start(self)**
  + field = Field()
  + field.prepare\_field()
  + self.player\_amount = self.view.set\_players(self.player\_range)
  + create\_players()
  + prepare\_player()
* **def create\_players(self)**
  + for player in range(0, self.player\_amount):
    - self.player\_list.append(Player())
* **def prepare\_player(self)**
  + \_\_counter = 0
  + for player in player\_list:
    - player.prepare\_player(counter)
    - counter += 1

class Field(Controller)

* **def \_\_init\_\_(self):**
  + *self.\_\_min\_length = 4*
  + *self.\_\_max\_length = 10*
  + *self.\_\_min\_height = 4*
  + *self.\_\_max\_height = 10*
  + *self.\_\_default\_length = 7*
  + *self.\_\_default\_height = 6*
  + *self.\_\_field = [7][6]*
  + *self.\_\_min\_field = [min\_length][min\_height]*
  + *self.\_\_max\_field = [max\_length][max\_height]*
  + *self.\_\_default\_field = [default\_length][default\_height]*
* **def prepare(self):**
  + self.field = self.view.prepare(self.min\_size, self.max\_size, self.default\_size)

class Player(Controller)

* **def \_\_init\_\_(self):**
  + \_\_turn = 0
  + *\_\_first\_name = “Player”*
  + *\_\_second\_name = “ 1”*
  + *\_\_token\_color = “Green”*
  + *\_\_score = 0*
* **def prepare\_player(self, turn):**
  + self.turn = turn
  + \_\_player\_information = self.view.prepare\_player()
  + first\_name = \_\_player\_information[0]
  + second\_name = \_\_player\_information[1]
  + token\_color = \_\_player\_information[2]

game = Controller(Tkinter OR Console OR PyQt5)

game.start()

File ConnectFourStrategy

Class View()

* **@staticmethod**
* **def prepare\_field():**
  + raise NotImplementedError
* **@staticmethod**
* **def set\_players():**
  + raise NotImplementedError
* **@staticmethod**
* **def prepare\_player():**
  + raise NotImplementedError

class Tkinter(View)

* **@staticmethod**
* **def prepare\_field():**
  + pass
* **@staticmethod**
* **def set\_players():**
  + pass
* **@staticmethod**
* **def prepare\_player():**
  + pass

class Console(View)

* **@staticmethod**
* **def prepare\_field(min\_size, max\_ size, default\_ size):**
  + \_\_field = default\_size if input(“Soll das Standardfeld verwendet werden? „).lower() == „ja“ else [0, 0]
  + while \_\_field[0] < min\_ size[0] or \_\_field[0] > max\_ size[0]:
    - \_\_field[0] = int(input(“Wie breit soll das Feld sein? “))
    - if \_\_field[0] < min\_size[0]:
      * print(„Das Feld ist zu schmal.“)
    - elif \_\_field[0] > max\_size[0]:
      * print(„Das Feld ist zu breit.“)
  + while \_\_field[0] < min\_size[0] or \_\_field[0] > max\_size[0]:
    - \_\_field[1] = int(input(“Wie hoch soll das Feld sein? “))
    - if \_\_field[1] < min\_size[1]:
      * print(„Das Feld ist zu tief.“)
    - elif \_\_field[1] > max\_size[1]:
      * print(„Das Feld ist zu hoch.“)
  + return \_\_field
* **@staticmethod**
* **def set\_players(players\_range)**
  + \_\_player\_amount = 2 if input(“Sollen zwei Spieler beteiligt sein? „).lower() == „ja“ else 0
  + while \_\_player\_amount < players\_range[0] or \_\_player\_amount > players\_range[1]:
    - \_\_player\_amount = int(input(“Wie viele Spieler soll es geben? “))
    - if \_\_player\_amount < players\_range[0]:
      * print(„Es gibt nicht genug Spieler.“)
    - elif \_\_player\_amount > players\_range[1]:
      * print(“Es gibt zu viele Spieler.“)
  + return \_\_player\_amount
* **@staticmethod**
* **def prepare\_player()**
  + \_\_first\_name = str(input(„Vorname: „)
  + \_\_second\_name = str(input(„Nachname: “)
  + \_\_token\_color = str(input(“Farbe: ”)
  + return [\_\_first\_name, \_\_second\_name, \_\_token\_color]

class PyQt5(View)

* **@staticmethod**
* **def prepare():**
  + pass
* **@staticmethod**
* **def set\_players():**
  + pass
* **@staticmethod**
* **def prepare\_player():**
  + pass