# Example 1. Guess-the-day ver1

|  |  |
| --- | --- |
| ver0 | original description |
| ver1 | finding class, attribute and method candidates |

## Original description

Select-the-day, the program shows that day’s weekday and three other randomly picked week days, asks a user to select which day is today, checks the answer and saves the information to data storage accessible to the personnel of the service home. If the user picked the wrong day, the program asks again giving a gentle hint e.g. today is the first day of the week. And if the user guesses wrong the second time, the program shows a nice greeting and sets the game as FAILED to the data storage. If the user answered correctly the second time the game is saved as FOLLOW, and if the answer was right the first time, the game is saved as PASSED. The personnel checks the saved games to verify if the resident has problems in keeping up with the time.

## 1st iteration, check nouns

**Select-the-day, the program shows that day’s weekday and three other randomly picked week days, asks a user to select which day is today, checks the answer and saves the information to data storage accessible to the personnel of the service home. If the user picked the wrong day, the program asks again giving a gentle hint e.g. today is the first day of the week. And if the user** **guesses wrong the second time, the program shows a nice greeting and sets the game as FAILED to the data storage. If the user answered correctly the second time the game is saved as FOLLOW, and if the answer was right the first time, the game is saved as PASSED. The personnel checks the saved games to verify if the resident has problems in keeping up with the time.**

|  |  |  |  |
| --- | --- | --- | --- |
| noun | basic form, synonym, duplicate value, notes | value or collections | possible class, object, attribute |
| **Select-the-day = Guess-the-day** |  | one value, name of the game, title | **Possible attribute:**  game’s title |
| **game** |  | object, one run of the game inside the program | **Possible class:**  knows the rules of the game and keeps track of the game’s data |
| **weekdays** | weekday | Object | **Possible object:**  Stores three randomly selected weekdays |
| **user** |  | object | **Possible class:** |
| **today** |  | object | **Possible attribute:**  Stores today’s weekday |
| **day** |  | object | **Possible attribute:**  Stores which day user anwered |
| **time** |  | Type integer | **Possible attribute:** Keeps count of user attempts |
| **answer** |  | Type boolean | **Possible attribute:**  Keeps value of user’s answers |
|  |  |  |  |

## 2nd iteration, check verbs

**Select-the-day, the program shows that day’s weekday and three other randomly picked week days, asks a user to select which day is today, checks the answer and saves the information to data storage accessible to the personnel of the service home. If the user picked the wrong day, the program asks again giving a gentle hint e.g. today is the first day of the week. And if the user** **guesses wrong the second time, the program shows a nice greeting and sets the game as FAILED to the data storage. If the user answered correctly the second time the game is saved as FOLLOW, and if the answer was right the first time, the game is saved as PASSED. The personnel checks the saved games to verify if the resident has problems in keeping up with the time.**

|  |  |  |
| --- | --- | --- |
| verb | subject – object (who does, who is the target) | possible action, function, method |
| **pick** | Game’s action | **Possible method:**  Randomly select three other weekdays |
| **select** | User’s action | **Possible method:**  Select which day is today |
| **check** | user’s action | **Possible method:**  Check is user’s answer correct |
| **save** | User’s action | **Possible method:**  Saves user’s answer to be accessed by service home personnel |
| **verify** | Game’s action | **Possible method:**  Personnel check user answers in later time |
| **ask** | Game’s action | **Possible method:**  Ask if wanted to retrieve recently created answers + show answers |
| **guess** | User’s action | **Possible method:**  Allow user to make guess of todays’ weekday |
| **show** | User’s action | **Possible method:**  Show 4 weekdays |
| **quit** | User’s action | **Possible method:**  Quits the game |
| **restart** | User’s action | **Possible method:**  Restart the game |

## 3rd iteration, specification ver1

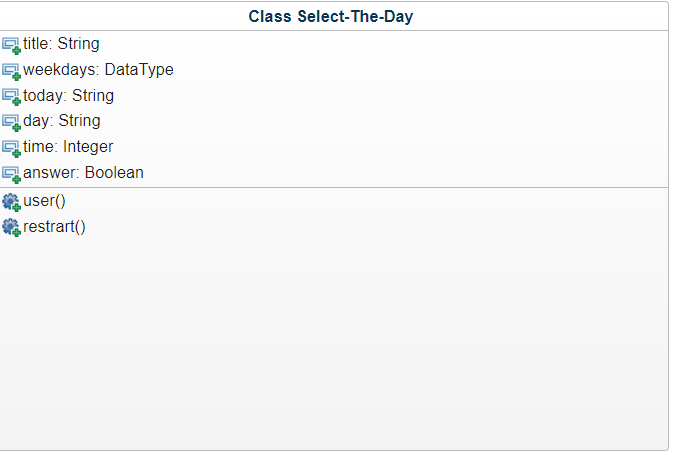
Select-the-day game asks user to select from 4 options which weekday is today. Game will save user answer as true or false. If first attempt fails, user is allowed to try again. If second attempt fails. Game ends and user gets message “fail”. If user answers correctly on one of the times, user gets message “passed”.

### Description

Ver 1

1. Guess-the-day is memory game
   1. Attributes
      1. Title
         1. Stores title of the game
      2. Today
         1. Stores today’s weekday
      3. Day
         1. Stores user’s answer which weekday user chose
      4. Time
         1. Keeps count of user’s attempts
      5. Answer
         1. Keeps value of user’s answers
   2. Methods
      1. Pick
         1. Randomly select three other weekdays
      2. Select
         1. Select which day is today
      3. Check
         1. Check is user’s answer correct
      4. Save
         1. Saves user’s answer to be accessed by service home personnel
      5. Verify
         1. Personnel check users answers in later time
      6. Ask
         1. Ask if wanted to retrieve recently created answers + show answers
      7. Guess
         1. Allow user to make guess of today’s weekday
      8. Show
         1. Show 4 weekdays. 3 randomly selected and 1 actual
      9. Quit
         1. Quits the game
      10. Restart
          1. Restart the game
   3. Classes
      1. Game
         1. knows the rules of the game and keeps track of the game’s data
            1. Handles game specific actions and data
      2. User
         1. Handles user interaction
   4. Reserve topic
   5. Reserve topic

### Generated UML class diagram (<https://app.genmymodel.com/>) @ver1



### Code generated from the UML diagram @ver1

class GuessTheDay:

def \_\_init\_\_(self):

self.title = ""

self.weekdays = None

self.today = ""

self.day = ""

self.time = 0

self.answer = False

# Start of user code -> properties/constructors for Class Select-The-Day class

# End of user code

def user(self):

# Start of user code protected zone for user function body

raise NotImplementedError

# End of user code

def restrart(self):

# Start of user code protected zone for restrart function body

raise NotImplementedError

# End of user code

# Start of user code -> methods for Class Select-The-Day class

# End of user code

# Start of user code -> functions/methods for memorygame package

# End of user code

Completed code ver 1

class GuessTheDay:

# Establishing parameters

def \_\_init\_\_(self):

self.title = "Select-the-day"

self.weekdays = ['monday', 'tuesday', 'wednesday', 'thursday', 'friday', 'saturday' 'sunday']

self.today = ""

self.day = ""

self.time = 0

self.answer = []

def game(self):

Pass

def user(self):

# Start of user code protected zone for user function body

pass

# End of user code

def restart(self):

# Start of user code protected zone for restrart function body

pass