

De La Salle University - Manila

**Term 3**,A.Y. **2019-2020**

In partial fulfillment

of the course

In **CCPROG3** (**S17**)

**Test Cases MCO2**

Submitted by:

**Avelino, Joris Gabriel L.**

**Gonzaga, Luis Miguel**

Submitted to:

**Ms. Katrina Solomon**

**September 28, 2020**

**Map Class**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Method** | **#** | **Description** | **Input** | **Expected Result** | **Actual Result** |
| Random Map selector | 1 | Random map that is equals to 1 | if ((map.nextInt(3) + 1) == 1) | It will select Map1 | Map1 is selected |
|  |  | Random map that is equals to 2 | if ((map.nextInt(3) + 1) == 2) | It will select Map2 | Map2 is selected |
|  |  | Random map that is equals to 3 | if ((map.nextInt(3) + 1) == 2) | It will select Map3 | Map3 is selected |
| printMap() | 2 | Map 1 is to be generated | printMap(currentMap); \*currentMap = Map1; | Map1 will be printed | Map1 is printed |
|  |  | Map 2 is to be generated | printMap(currentMap); \*currentMap = Map2; | Map2 will be printed | Map2 is printed |
|  |  | Map 3 is to be generated | printMap(currentMap); \*currentMap = Map3; | Map3 will be printed | Map3 is printed |
| pos.equalsIgnoreCase(Down) | 3 | User inputs in lowercase | down | Input will be accepted | Input is accepted |
|  |  | User inputs in random case | dOwn | Input will be accepted | Input is accepted |
|  |  | User inputs in uppercase | DOWN | Input will be accepted | Input is accepted |
| pos.equalsIgnoreCase(Up) | 4 | User inputs in lowercase | up | Input will be accepted | Input is accepted |
|  |  | User inputs in random case | uP | Input will be accepted | Input is accepted |
|  |  | User inputs in uppercase | UP | Input will be accepted | Input is accepted |
| pos.equalsIgnoreCase(Left) | 5 | User inputs in lowercase | left | Input will be accepted | Input is accepted |
|  |  | User inputs in random case | leFT | Input will be accepted | Input is accepted |
|  |  | User inputs in uppercase | LEFT | Input will be accepted | Input is accepted |
| pos.equalsIgnoreCase(Right) | 6 | User inputs in lowercase | right | Input will be accepted | Input is accepted |
|  |  | User inputs in random case | rIGht | Input will be accepted | Input is accepted |
|  |  | User inputs in uppercase | RIGHT | Input will be accepted | Input is accepted |
| Invalid Input | 7 | User inputs invalid direction | sideways | Input will not be accepted | "Invalid direction!"  Is displayed |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Method** | **#** | **Test Description** | **Sample Data** | **Expected Output** | **Actual Output** |
| public LogInController | 1 | Initiates the controller class for the logInController | LogInView, player | Working | working |
|  | 2 | Not enough parameters inputted | Player | Error | Error |
|  | 3 | Parameters are not in order | Player, LogInView | Error | Error |
| void chooseCompanionSequence | 4 | initiates the choosing of companion sequence | Player | GUI will appear | GUI will appear |
|  | 5 | Different parameter was inputted | Companion | GUI will not appear | GUI will not appear |
|  | 6 | No parameter was inputted |  | GUI will not appear | GUI will not appear |
| class CreateButtonListener void actionPerformed |  | User inputted wrong avatar preference | “girl” | Ask player again | Ask player again |
|  |  | User inputted right avatar preference | “Female” | Create player |  |
|  |  | User inputted right string but in lowercase | “female” | Create player |  |

# LogInController Class

# LogInView

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Method** | **#** | **Test Description** | **Sample Data** | **Expected Output** | **Actual Output** |
| public LogInView | 1 | Initiates the controller class for the LogInView |  | Creates the GUI | Creates the GUI |
| void createButtonListener | 2 | Creates an ActionListener to make the button work | ActionListener | Makes the button work | Makes the button work |
|  | 3 | Inputs a int as a parameter | 1 | Button will not work | Button will not work |
|  | 4 | Inputs a string as a parameter | “class” | Button will not work | Button will not work |
| void displayErrorMessage | 5 | Inputs a string | “error” | Show a string in the message dialogue | Show a string in the message dialogue |
|  | 6 | Inputs an int | 1 | Show an int in the message dialogue | Show an int in the message dialogue |
|  | 7 | No input message |  | Error | Error |
| void setSuccessLabel | 8 | Inputs a string | “working” | Show “working” in the sucessLabel | Show “working” in the sucessLabel |
|  | 9 | Inputs an int | 2 | Error | Error |
|  | 10 | No input |  | Error | Error |

# ChooseCompanionView Class

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Method** | **#** | **Test Description** | **Sample Data** | **Expected Output** | **Actual Output** |
| public ChooseCompanionView | 1 | Calls the GUI |  | Presents the GUI |  |
| Void initialize | 2 | Initializes/creates the GUI |  | Creates the GUI |  |
| void kirinImageListener | 3 | Creates an ActionListener to make the button work | ActionListener | Makes the button work | Makes the button work |
|  | 4 | Inputs a int as a parameter | 1 | Button will not work | Button will not work |
|  | 5 | Inputs a string as a parameter | “class” | Button will not work | Button will not work |
| void yukiImageListener | 6 | Creates an ActionListener to make the button work | ActionListener | Makes the button work | Makes the button work |
|  | 7 | Inputs a int as a parameter | 1 | Button will not work | Button will not work |
|  | 8 | Inputs a string as a parameter | “class” | Button will not work | Button will not work |
| void yumeImageListener | 9 | Creates an ActionListener to make the button work | ActionListener | Makes the button work | Makes the button work |
|  | 10 | Inputs a int as a parameter | 1 | Button will not work | Button will not work |
|  | 11 | Inputs a string as a parameter | “class” | Button will not work | Button will not work |
| void sameImageListener | 12 | Creates an ActionListener to make the button work | ActionListener | Makes the button work | Makes the button work |
|  | 13 | Inputs a int as a parameter | 1 | Button will not work | Button will not work |
|  | 14 | Inputs a string as a parameter | “class” | Button will not work | Button will not work |
| void kirinYesListener | 15 | Creates an ActionListener to make the button work | ActionListener | Makes the button work | Makes the button work |
|  | 16 | Inputs a int as a parameter | 1 | Button will not work | Button will not work |
|  | 17 | Inputs a string as a parameter | “class” | Button will not work | Button will not work |
| void kirinYesListener | 18 | Creates an ActionListener to make the button work | ActionListener | Makes the button work | Makes the button work |
|  | 19 | Inputs a int as a parameter | 1 | Button will not work | Button will not work |
|  | 20 | Inputs a string as a parameter | “class” | Button will not work | Button will not work |
| void yukiYesListener | 21 | Creates an ActionListener to make the button work | ActionListener | Makes the button work | Makes the button work |
|  | 22 | Inputs a int as a parameter | 1 | Button will not work | Button will not work |
|  | 23 | Inputs a string as a parameter | “class” | Button will not work | Button will not work |
| void yumeYesListener | 24 | Creates an ActionListener to make the button work | ActionListener | Makes the button work | Makes the button work |
|  | 25 | Inputs a int as a parameter | 1 | Button will not work | Button will not work |
|  | 26 | Inputs a string as a parameter | “class” | Button will not work | Button will not work |
| void sameYesListener | 27 | Creates an ActionListener to make the button work | ActionListener | Makes the button work | Makes the button work |
|  | 28 | Inputs a int as a parameter | 1 | Button will not work | Button will not work |
|  | 29 | Inputs a string as a parameter | “class” | Button will not work | Button will not work |

# ChooseCompanionController Class

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Method** | **#** | **Test Description** | **Sample Data** | **Expected Output** | **Actual Output** |
| public ChooseCompanionController | 1 | Initiates the controller | ChooseCompanionView, player | Working | Working |
|  | 2 | Not enough parameters inputted | Player | Error | Error |
|  | 3 | Parameters are not in order | Player, ChooseCompanionView | Error | Error |
| void addCompanionList | 4 | Inputs a companion | K | Added to Companion initial to List of chosen companions | Added to Companion initial to List of chosen companions |
|  | 5 | Inputs a character | Kirin | Error | Error |
|  | 6 | Inputs a lowercase initial | K | Error | Error |
| boolean checkIfCompanionsThree | 7 | Inputs an int less than 3 | 2 | False | False |
|  | 8 | Inputs an int equals to 3 | 3 | True | True |
|  | 9 | Inputs an int more than 3 | 4 | True | True |
| void addPanelChooseCompanion | 10 | Inputs a correct String | “kirinPanel” | Adds the panel | Adds the panel |
|  | 11 | Inputs an incorrect String | “kPanel” | Panel will be removed | Panel will be removed |
|  | 12 | Inputs a character | ‘k’ | Panel will be removed | Panel will be removed |
| class KirinPanelListener  void actionPerformed | 13 | Button is clicked |  | kirinPanel will be added | kirinPanel will be added |
| class YumePanelListener  void actionPerformed | 14 | Button is clicked |  | yumePanel will be added | yumePanel will be added |
| class YukiPanelListener  void actionPerformed | 15 | Button is clicked |  | yukiPanel will be added | yukiPanel will be added |
| class SamePanelListener  void actionPerformed | 16 | Button is clicked |  | samePanel will be added | samePanel will be added |
| void checkCompleteCompanions | 17 | Player had picked 3 companions |  | Show message dialogue | Show message dialogue |
|  | 18 | Player had picked 2 companions |  | No message dialogue shown | No message dialogue shown |
|  | 19 | Player had picked 0 companions |  | No message dialogue shown | No message dialogue shown |
| void choosingCompanionList | 20 | Adds a panel with a right parameter passed | ‘K’ | Show the right panel | Show the right panel |
|  | 21 | Don’t show anything because the parameter passed was wrong | ‘A’ | Show nothing | Show nothing |
|  | 22 | Input an int instead of a char | 1 | Show nothing | Show nothing |
| class KirinButtonListener  void actionPerformed | 23 | User clicked the button and will add that companion to their list of companions | ‘K’ | Add companion to their list of companions | Add companion to their list of companions |
| class SameButtonListener  void actionPerformed | 24 | User clicked the button and will add that companion to their list of companions | ‘S’ | Add companion to their list of companions | Add companion to their list of companions |
| class YumeButtonListener  void actionPerformed | 25 | User clicked the button and will add that companion to their list of companions | ‘Y’ | Add companion to their list of companions | Add companion to their list of companions |
| class YukiButtonListener  void actionPerformed | 26 | User clicked the button and will add that companion to their list of companions | ‘I’ | Add companion to their list of companions | Add companion to their list of companions |

# EncounterView Class

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Method** | **#** | **Test Description** | **Sample Data** | **Expected Output** | **Actual Output** |
| public Encounter | 1 | Initializes/creates the GUI |  | Shows the GUI | Shows the GUI |
| void chooseCompanionListener | 2 | Creates an ActionListener to make the button work | ActionListener | Makes the button work | Makes the button work |
|  | 3 | Inputs a int as a parameter | 1 | Button will not work | Button will not work |
|  | 4 | Inputs a string as a parameter | “class” | Button will not work | Button will not work |
| void pickAgainButtonListener | 5 | Creates an ActionListener to make the button work | ActionListener | Makes the button work | Makes the button work |
|  | 6 | Inputs a int as a parameter | 1 | Button will not work | Button will not work |
|  | 7 | Inputs a string as a parameter | “class” | Button will not work | Button will not work |
| void buttonCompanion1Listener | 8 | Creates an ActionListener to make the button work | ActionListener | Makes the button work | Makes the button work |
|  | 9 | Inputs a int as a parameter | 1 | Button will not work | Button will not work |
|  | 10 | Inputs a string as a parameter | “class” | Button will not work | Button will not work |
| void buttonCompanion2Listener | 11 | Creates an ActionListener to make the button work | ActionListener | Makes the button work | Makes the button work |
|  | 12 | Inputs a int as a parameter | 1 | Button will not work | Button will not work |
|  | 13 | Inputs a string as a parameter | “class” | Button will not work | Button will not work |
| void buttonCompanion3Listener | 14 | Creates an ActionListener to make the button work | ActionListener | Makes the button work | Makes the button work |
|  | 15 | Inputs a int as a parameter | 1 | Button will not work | Button will not work |
|  | 16 | Inputs a string as a parameter | “class” | Button will not work | Button will not work |
| void confirmButtonListener | 17 | Creates an ActionListener to make the button work | ActionListener | Makes the button work | Makes the button work |
|  | 18 | Inputs a int as a parameter | 1 | Button will not work | Button will not work |
|  | 19 | Inputs a string as a parameter | “class” | Button will not work | Button will not work |
| void attackNextButtonListener | 20 | Creates an ActionListener to make the button work | ActionListener | Makes the button work | Makes the button work |
|  | 21 | Inputs a int as a parameter | 1 | Button will not work | Button will not work |
| void buttonNextListener | 22 | Inputs a string as a parameter | “class” | Button will not work | Button will not work |

# EncounterController Class

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Method** | **#** | **Test Description** | **Sample Data** | **Expected Output** | **Actual Output** |
| public Encounter | 1 | Initiates the controller | ChooseCompanionView, player, enemy | Working | Working |
|  | 2 | Not enough parameters inputted | Player | Error | Error |
|  | 3 | Parameters are not in order | Player, ChooseCompanionView, enemy | Error | Error |
| void initiatePlayerCompanionButtons | 4 | Set Player companion in the buttons |  | Show companion buttons with correct names | Show companion buttons with correct names |
| void checkCompanionHealth | 5 | Companion is alive | Health = 50 | Useable in an encounter | Useable in an encounter |
|  | 6 | Companion is dead | Health = 0 | Can’t be used in encounter | Can’t be used in encounter |
| void damageDescription | 7 | Companion dealt a lot of damage | Damage = 10 | Displays damage description | Displays damage description |
|  | 8 | Companion dealt low damage | Damage = 2 | Displays damage description | Displays damage description |
|  | 9 | Companion dealt no damage | Damage = 0 | Displays damage description | Displays damage description |
| void setCompanionStats | 10 | Chose right companion for encounter | Kirin | Displays companion’s stats | Displays companion’s stats |
| void addPanelEncounter | 11 | Inputs a correct String | “kirinPanel” | Adds the panel | Adds the panel |
|  | 12 | Inputs an incorrect String | “kPanel” | Panel will be removed | Panel will be removed |
|  | 13 | Inputs a character | ‘k’ | Panel will be removed | Panel will be removed |
| class ChooseCompanionListener  void actionPerformed | 14 | Button is clicked |  | listCompanionsDescriptionwill be added | listCompanionsDescriptionwill be added |
| class ButtonCompanion1Listener  void actionPerformed | 15 | Button is clicked |  | Shows the companionStatsPanel and the stats of the selected companion | Shows the companionStatsPanel and the stats of the selected companion |
| class ButtonCompanion2Listener  void actionPerformed | 16 | Button is clicked |  | Shows the companionStatsPanel and the stats of the selected companion | Shows the companionStatsPanel and the stats of the selected companion |
| class ButtonCompanion3Listener  void actionPerformed | 17 | Button is clicked |  | Shows the companionStatsPanel and the stats of the selected companion | Shows the companionStatsPanel and the stats of the selected companion |
| class ConfirmButtonListener  void actionPerformed | 18 | Button is clicked |  | Shows the attackPanel and the stats of the selected companion | Shows the attackPanel and the stats of the selected companion |
| class AttackNextButtonListener  void actionPerformed | 19 | Button is clicked |  | Shows the dealt damage with the description | Shows the dealt damage with the description |
| class ButtonNextListener  void actionPerformed | 20 | Button is clicked |  | Shows who won or whose turn is it next | Shows who won or whose turn is it next |

References:

Kirin

Akulich, V., 2015. *Creatures And Cute Monsters*. [image] Available at: <https://www.behance.net/gallery/27669265/Creatures-and-cute-monsters>

Same

Dragomir, R. (n.d.). *Celtic Dragon Drawing* [image]. Available at: <https://www.pinterest.ph/pin/393572454909574813/>

Yume

Joamaa. (2018, April 14). *Cute cloud creature*. Retrieved from https://drawception.com/panel/drawing/01dA6336/cute-cloud-creature/ (Originally photographed 2018, April 14)

Yuki

Bulmahn, J. (n.d.). *Earth elemental*. Retrieved from https://pathfinderwiki.com/wiki/Earth\_elemental (Originally photographed 2009)

Fireworks

Deck, C. (n.d.). *Fireworks gif*. Retrieved from <https://www.pinterest.ph/pin/686376799430735006/?autologin=true&nic_v2=1a7baYdmF>

*Brick Wall*. (n.d.). Retrieved September 28, 2020, from

%3A%2F%2Fwww.popsci.com%2Fhistory-bricks-game-thrones%2F&h=AT3Mp5WwsrbJDBOknqqtJjkERlYDSLI-sOSuzs5r9ZvY-iYedxsTzVCyCQIQY34etnUBC-vmHPf03kclTRoD2rBNbvOPnG1Oraw4s1PINrWFF7Fou0QZtZbsBSHBXg1V6yDQD2BTWdaHS9AiKt8

*Checkerboard*. (n.d.). Retrieved September 28, 2020, from https://l.messenger.com/l.php?u=https%3A%2F%2Fmidsouthcraftingsupplies.com%2F14-12-checkerboard-fashion-print-adhesive&h=AT3Mp5WwsrbJDBOknqqtJjkERlYDSLI-sOSuzs5r9ZvY-iYedxsTzVCyCQIQY34etnUBC-vmHPf03kclTRoD2rBNbvOPnG1Oraw4s1PINrWFF7Fou0QZtZbsBSHBXg1V6yDQD2BTWdaHS9AiKt8

*Heart Panel*. (n.d.). Retrieved September 28, 2020, from   
https%3A%2F%2Fwallpaperset.com%2Fcute-heart-wallpaper&h=AT3Mp5WwsrbJDBOknqqtJjkERlYDSLI-sOSuzs5r9ZvY-iYedxsTzVCyCQIQY34etnUBC-vmHPf03kclTRoD2rBNbvOPnG1Oraw4s1PINrWFF7Fou0QZtZbsBSHBXg1V6yDQD2BTWdaHS9AiKt8

*Mario*. (n.d.). Retrieved September 28, 2020, from https%3A%2F%2Fvignette.wikia.nocookie.net%2Fnintendo%2Fimages%2Fd%2Fd8%2FNew\_Super\_Mario\_Bros.\_U\_Deluxe\_-\_Mario\_01.png%2Frevision%2Flatest%3Fcb%3D20181226204245%26path-prefix%3Den&h=AT3Mp5WwsrbJDBOknqqtJjkERlYDSLI-sOSuzs5r9ZvY-iYedxsTzVCyCQIQY34etnUBC-vmHPf03kclTRoD2rBNbvOPnG1Oraw4s1PINrWFF7Fou0QZtZbsBSHBXg1V6yDQD2BTWdaHS9AiKt8

*Princess Peach*. (n.d.). Retrieved September 28, 2020, from https%3A%2F%2Fstatic.wikia.nocookie.net%2Floveinterest%2Fimages%2Fb%2Fb5%2FPeach\_-\_Mario\_Party\_10.png%2Frevision%2Flatest%3Fcb%3D20160110012048&h=AT3Mp5WwsrbJDBOknqqtJjkERlYDSLI-sOSuzs5r9ZvY-iYedxsTzVCyCQIQY34etnUBC-vmHPf03kclTRoD2rBNbvOPnG1Oraw4s1PINrWFF7Fou0QZtZbsBSHBXg1V6yDQD2BTWdaHS9AiKt8