**Records and Files** **Spot Check Electronic Answer Document (EAD)**

Use the following document to record your answers to the Lists spot check. You should then submit the completed EAD to the link provided on Moodle by your teacher.

|  |
| --- |
| **Question 1, Part d** |
| def load\_games(filename):  pass  def save\_games(filename, games):  pass  #the parameter is games because eventually you will be displaying  #multiple games using this function  class database():  def \_\_init\_\_(self):  self.name = None  self.platform = None  self.genre = None  self.cost = None  self.no\_of\_players = None  self.online\_functionality = None  def display\_games(games):  print("-" \* 105)  print("|Name: Platform: Genre: Cost: Number of players: Online functionality? |")  for game in games:  print("-" \* 105)  print("|{0:<12} {1:<15} {2:<12} {3:<15} {4:<25} {5:<15} |".format(game.name, game.platform, game.genre, game.cost, game.no\_of\_players, game.online\_functionality))  print("-" \* 105)  pass  def get\_game\_from\_user():  games = []  game = database()  game.name = input("Please input the name of the game: ")  game.platform = input("Please input the platform the game is on: ")  game.genre = input("Please input the genre of the game: ")  game.cost = input("Please input the cost of the game: ")  game.no\_of\_players = int(input("Please input the number of players: "))  game.online\_functionality = input("Please state whether the game has onlne functionality or not: ")  games.append(game)  return games  pass  def display\_menu():  print()  print("\*\*\*Welcome to the Computer and Video Game Database\*\*\*")  print()  print("1. Add new games")  print("2. Display games")  print("3. Exit program")  print()  def main():  exit\_program = False  while not exit\_program:  display\_menu()  selected\_option = int(input("Please select a menu option: "))  if selected\_option == 1:  games = get\_game\_from\_user()  pass  elif selected\_option == 2:  display\_games(games)  pass  elif selected\_option == 3:  pass  else:  print("Please enter a valid option (1-3)")  print()  if \_\_name\_\_ == "\_\_main\_\_":  main() |
| **Question 1, Part e** |
|  |

|  |
| --- |
| **Question 2, Part d** |
| def load\_games(filename):  pass  def save\_games(filename, games):  pass  #the parameter is games because eventually you will be displaying  #multiple games using this function  class database():  def \_\_init\_\_(self):  self.name = None  self.platform = None  self.genre = None  self.cost = None  self.no\_of\_players = None  self.online\_functionality = None  def display\_games(games):  print("-" \* 105)  print("|Name: Platform: Genre: Cost: Number of players: Online functionality? |")  for game in games:  print("-" \* 105)  print("|{0:<12} {1:<15} {2:<12} {3:<15} {4:<25} {5:<15} |".format(game.name, game.platform, game.genre, game.cost, game.no\_of\_players, game.online\_functionality))  print("-" \* 105)  pass  def get\_game\_from\_user():  game = database()  games = []  check = ""  while check != "-1":  check = input("Enter -1 to input the final record, or return to continue: ")  game.name = input("Please input the name of the game: ")  game.platform = input("Please input the platform the game is on: ")  game.genre = input("Please input the genre of the game: ")  game.cost = input("Please input the cost of the game: ")  game.no\_of\_players = int(input("Please input the number of players: "))  game.online\_functionality = input("Please state whether the game has online functionality or not: ")  games.append(game)  return games  pass  def display\_menu():  print()  print("\*\*\*Welcome to the Computer and Video Game Database\*\*\*")  print()  print("1. Add new games")  print("2. Display games")  print("3. Exit program")  print()  def main():  exit\_program = False  while not exit\_program:  display\_menu()  selected\_option = int(input("Please select a menu option: "))  if selected\_option == 1:  games = get\_game\_from\_user()  pass  elif selected\_option == 2:  display\_games(games)  pass  elif selected\_option == 3:  pass  else:  print("Please enter a valid option (1-3)")  print()  if \_\_name\_\_ == "\_\_main\_\_":  main() |
| **Question 2, Part e** |
|  |

|  |
| --- |
| **Question 3, Part c** |
| def save\_games(games):  with open("save\_game.txt", mode = "w", encoding = "utf-8") as save\_game:  for game in games:  save\_game.write(game.name)  save\_game.write(game.platform)  save\_game.write(game.genre)  save\_game.write(game.cost)  save\_game.write(game.no\_of\_players)  save\_game.write(game.online\_functionality)  pass  def main():  exit\_program = False  while not exit\_program:  display\_menu()  selected\_option = int(input("Please select a menu option: "))  if selected\_option == 1:  games = get\_game\_from\_user()  pass  elif selected\_option == 2:  display\_games(games)  pass  elif selected\_option == 3:  save\_games(games)  exit()  pass  else:  print("Please enter a valid option (1-3)")  print() |

|  |
| --- |
| **Question 4, Part c** |
| def load\_games():  with open ("save\_game.txt", mode = "r", encoding = "utf-8") as save\_game:  for game in games:  save\_game.read(game.name)  save\_game.read(game.platform)  save\_game.read(game.genre)  save\_game.read(game.cost)  save\_game.read(game.no\_of\_players)  save\_game.read(game.online\_functionality)  pass  def main():  exit\_program = False  load\_games()  while not exit\_program:  display\_menu()  selected\_option = int(input("Please select a menu option: "))  if selected\_option == 1:  games = get\_game\_from\_user()  pass  elif selected\_option == 2:  display\_games(games)  pass  elif selected\_option == 3:  save\_games(games)  exit()  pass  else:  print("Please enter a valid option (1-3)")  print() |
| **Question 4, Part d** |
|  |
| **Question 4, Part e** |
|  |

|  |
| --- |
| **Question 5, Part a** |
| It will show a value error because the data type is invalid if not an integer. |
| **Question 5, Part c** |
|  |
| **Question 5, Part d** |
|  |