

# PPL Assignment - Question9

API Documentation

April 10, 2017

## Contents

<b>Contents</b>	<b>1</b>
<b>1 Module Boys</b>	<b>2</b>
1.1 Variables . . . . .	2
1.2 Class Boy . . . . .	2
1.2.1 Methods . . . . .	2
<b>2 Module Gifts</b>	<b>3</b>
2.1 Variables . . . . .	3
2.2 Class Gift . . . . .	3
2.2.1 Methods . . . . .	3
<b>3 Module Girls</b>	<b>4</b>
3.1 Variables . . . . .	4
3.2 Class Girl . . . . .	4
3.2.1 Methods . . . . .	4
<b>4 Module boyGeek</b>	<b>5</b>
4.1 Variables . . . . .	5
4.2 Class boyGeek . . . . .	5
4.2.1 Methods . . . . .	5
<b>5 Module boyGenerous</b>	<b>6</b>
5.1 Variables . . . . .	6
5.2 Class boyGenerous . . . . .	6
5.2.1 Methods . . . . .	6
<b>6 Module boyMiser</b>	<b>7</b>
6.1 Variables . . . . .	7
6.2 Class boyMiser . . . . .	7
6.2.1 Methods . . . . .	7
<b>7 Module chooseFromBest</b>	<b>8</b>
7.1 Variables . . . . .	8
7.2 Class formCouples . . . . .	8
7.2.1 Methods . . . . .	8
<b>8 Module couple</b>	<b>9</b>

8.1	Variables . . . . .	9
8.2	Class couple . . . . .	9
8.2.1	Methods . . . . .	9
<b>9</b>	<b>Module driver</b>	<b>10</b>
9.1	Functions . . . . .	10
9.2	Variables . . . . .	10
<b>10</b>	<b>Module generator</b>	<b>11</b>
10.1	Variables . . . . .	11
10.2	Class randomGenerator . . . . .	11
10.2.1	Methods . . . . .	11
<b>11</b>	<b>Module giftEssential</b>	<b>12</b>
11.1	Variables . . . . .	12
11.2	Class giftEssential . . . . .	12
11.2.1	Methods . . . . .	12
<b>12</b>	<b>Module giftLuxury</b>	<b>13</b>
12.1	Variables . . . . .	13
12.2	Class giftLuxury . . . . .	13
12.2.1	Methods . . . . .	13
<b>13</b>	<b>Module giftUtility</b>	<b>14</b>
13.1	Variables . . . . .	14
13.2	Class giftUtility . . . . .	14
13.2.1	Methods . . . . .	14
<b>14</b>	<b>Module girlChoosy</b>	<b>15</b>
14.1	Variables . . . . .	15
14.2	Class girlChoosy . . . . .	15
14.2.1	Methods . . . . .	15
<b>15</b>	<b>Module girlDesperate</b>	<b>16</b>
15.1	Variables . . . . .	16
15.2	Class girlDesperate . . . . .	16
15.2.1	Methods . . . . .	16
<b>16</b>	<b>Module girlNormal</b>	<b>17</b>
16.1	Variables . . . . .	17
16.2	Class girlNormal . . . . .	17
16.2.1	Methods . . . . .	17
<b>17</b>	<b>Module q9</b>	<b>18</b>
17.1	Functions . . . . .	18
<b>18</b>	<b>Module reader</b>	<b>19</b>
18.1	Variables . . . . .	19
18.2	Class read . . . . .	19
18.2.1	Methods . . . . .	19
<b>19</b>	<b>Script script-girls_csv</b>	<b>20</b>

---

<b>20 Script script-logGifting.txt</b>	<b>21</b>
20.1 Variables . . . . .	21

# 1 Module Boys

## 1.1 Variables

Name	Description
--package--	<b>Value:</b> None

## 1.2 Class Boy

**Known Subclasses:** boyGeek.boyGeek, boyGenerous.boyGenerous, boyMiser.boyMiser

The parent class for all types of boys

### 1.2.1 Methods

**`__init__(self, boy)`**

constructor which initializes attributes such as :- name = name of the boy attractiveness = attractiveness of the boy intelligence = intelligence of the boy budget = budget of the boy minimumAttrReq = minimum attractiveness required in the girl bType = type of the boy status = relationship status

**`gifting(self)`**

**`happinessCalculator(self)`**

## 2 Module Gifts

### 2.1 Variables

Name	Description
--package--	<b>Value:</b> None

### 2.2 Class Gift

**Known Subclasses:** giftEssential.giftEssential, giftLuxury.giftLuxury, giftUtility.giftUtility

Parent class for all types of Gifts

#### 2.2.1 Methods

<code>--init--(self, gift)</code>
constructor with attributes :- giftType = type of the gift price = price of the gift value = value of the gift

### 3 Module Girls

#### 3.1 Variables

Name	Description
--package--	<b>Value:</b> None

#### 3.2 Class Girl

**Known Subclasses:** girlChoosy.girlChoosy, girlDesperate.girlDesperate, girlNormal.girlNormal

”Parent class for all types of girls

##### 3.2.1 Methods

**`__init__(self, girl)`**

constructor which initializes attributes such as :- name = name of the girl attractiveness = attractiveness of the girl intelligence = intelligence of the girl maintainanceCost = maintainance cost of the girl chosingCri = chosing criterion for the boys gType = type of the girl status = relationship status

**`happinessCalculator(self)`**

## 4 Module boyGeek

### 4.1 Variables

Name	Description
<code>--package--</code>	<b>Value:</b> None

### 4.2 Class boyGeek

Boys.Boy └─  
          boyGeek.boyGeek

Boy class for boyType = 'Geek'

#### 4.2.1 Methods

**`__init__(self, boy)`**

constructor , calls the parent constructor and initializes other attributes as:- happiness = happiness of the boy amountSpent = amount spent on gifting gfName = name of the girlfriend

Overrides: Boys.Boy.\_\_init\_\_

**`happinessCalculator(self, gIntelligence)`**

Calculates happiness for Geek boys

Overrides: Boys.Boy.happinessCalculator

**`gifting(self, gMaintenanceCost, Gifts, giftBasket)`**

Sets up the gift basket for Miser boys

Overrides: Boys.Boy.gifting

## 5 Module *boyGenerous*

### 5.1 Variables

Name	Description
<code>--package--</code>	<b>Value:</b> None

### 5.2 Class *boyGenerous*

Boys.Boy └─ **boyGenerous.boyGenerous**

Boy class for boyType = 'Generous'

#### 5.2.1 Methods

**`__init__(self, boy)`**

constructor , calls the parent constructor and initializes other attributes as:- happiness = happiness of the boy amountSpent = amount spent on gifting gfName = name of the girlfriend

Overrides: Boys.Boy.`__init__`

**`happinessCalculator(self, gHappiness)`**

Calculates happiness for Generous boys

Overrides: Boys.Boy.`happinessCalculator`

**`gifting(self, gMaintenanceCost, Gifts, giftBasket)`**

Sets up the gift basket for Generous boys

Overrides: Boys.Boy.`gifting`



## 6 Module boyMiser

### 6.1 Variables

Name	Description
<code>--package--</code>	<b>Value:</b> None

### 6.2 Class boyMiser

Boys.Boy └─  
          boyMiser.boyMiser

Boy class for boyType = 'Miser'

#### 6.2.1 Methods

**`__init__(self, boy)`**

constructor , calls the parent constructor and initializes other attributes as:- happiness = happiness of the boy amountSpent = amount spent on gifting gfName = name of the girlfriend

Overrides: Boys.Boy.\_\_init\_\_

**`gifting(self, gMaintainanceCost, Gifts, giftBasket)`**

Sets up the gift basket for Miser boys

Overrides: Boys.Boy.gifting

**`happinessCalculator(self)`**

Calculates happiness for Miser boys

Overrides: Boys.Boy.happinessCalculator

## 7 Module *chooseFromBest*

### 7.1 Variables

Name	Description
Couples	<b>Value:</b> []
Gifts	<b>Value:</b> []
--package--	<b>Value:</b> None
arrBoys	<b>Value:</b> []
arrGirls	<b>Value:</b> []

### 7.2 Class *formCouples*

#### 7.2.1 Methods

**kBestItems**(*self*, *k*, *list*, *criteria*)

the function creates a list of the best k items based on a particular criteria

**makeCouples**(*self*, *k*)

## 8 Module couple

### 8.1 Variables

Name	Description
<code>--package--</code>	<b>Value:</b> None

### 8.2 Class couple

Couple class with attributes:- bName = Name of the boy gName = Name of the girl happiness = happiness of the couple compatibility = compatibility of the couple

#### 8.2.1 Methods

<code>--init--(self, boy, girl)</code>
constructor

<code>happinessCalcuator(self, boy, girl)</code>
calculates happiness of the couple

<code>compatibilityCalculator(self, boy, girl)</code>
calculates compatibility of the couple

## 9 Module driver

### 9.1 Functions

<b>generateRandomInput()</b>
Can be used to generate random Inputs

### 9.2 Variables

Name	Description
--package--	<b>Value:</b> None

## 10 Module generator

### 10.1 Variables

Name	Description
<code>--package--</code>	<b>Value:</b> None

### 10.2 Class *randomGenerator*

#### 10.2.1 Methods

<code>--init--(<i>self</i>, <i>totalBoys</i>, <i>totalGirls</i>, <i>totalGifts</i>)</code>
--

<code>generateBoys(<i>self</i>)</code>
--

<code>generateGirls(<i>self</i>)</code>
---

<code>generateGifts(<i>self</i>)</code>
---

## 11 Module giftEssential

### 11.1 Variables

Name	Description
<code>--package--</code>	<b>Value:</b> None

### 11.2 Class giftEssential

Gifts.Gift —  
                   **giftEssential.giftEssential**

Gift Class for gifttype = 'Essential'

#### 11.2.1 Methods

<code>__init__(self, gift)</code>
constructor
Overrides: Gifts.Gift.__init__

## 12 Module **giftLuxury**

### 12.1 Variables

Name	Description
<code>--package--</code>	<b>Value:</b> None

### 12.2 Class **giftLuxury**

Gifts.Gift —  
**giftLuxury.giftLuxury**

Gift Class for gifttype = 'Luxury'

#### 12.2.1 Methods

<code>__init__(self, gift)</code>
constructor
Overrides: Gifts.Gift.__init__

## 13 Module `giftUtility`

### 13.1 Variables

Name	Description
<code>--package--</code>	<b>Value:</b> None

### 13.2 Class `giftUtility`

Gifts.Gift —  
                   **giftUtility.giftUtility**

Gift Class for gifttype = 'Utility'

#### 13.2.1 Methods

<b><code>__init__(self, gift)</code></b>
constructor
Overrides: Gifts.Gift. <code>__init__</code>



## 14 Module *girlChoosy*

### 14.1 Variables

Name	Description
<code>--package--</code>	<b>Value:</b> None

### 14.2 Class *girlChoosy*

Girls.Girl —  
                     *girlChoosy.girlChoosy*

Girl class for `girlType = 'Choosy'`

#### 14.2.1 Methods

<b><code>__init__(self, girl)</code></b>
constructor
Overrides: Girls.Girl. <code>__init__</code>

<b><code>happinessCalculator(self, giftBasket, amount)</code></b>
Calculates the happiness for girls of type Choosy
Overrides: Girls.Girl. <code>happinessCalculator</code>

## 15 Module *girlDesperate*

### 15.1 Variables

Name	Description
<code>--package--</code>	<b>Value:</b> None

### 15.2 Class *girlDesperate*

Girls.Girl —  
                     *girlDesperate.girlDesperate*

Girl class for girlType = 'Choosy'

#### 15.2.1 Methods

<b><code>__init__(self, girl)</code></b>
constructor
Overrides: Girls.Girl.__init__

<b><code>happinessCalculator(self, giftBasket, amount)</code></b>
Calculates happiness for girls of type Normal
Overrides: Girls.Girl.happinessCalculator

## 16 Module *girlNormal*

### 16.1 Variables

Name	Description
<code>--package--</code>	<b>Value:</b> None

### 16.2 Class *girlNormal*

Girls.Girl —  
                   **girlNormal.girlNormal**

Girl class for `girlType = 'Choosy'`

#### 16.2.1 Methods

<b><code>__init__(self, girl)</code></b>
constructor
Overrides: Girls.Girl.__init__

<b><code>happinessCalculator(self, giftBasket, amount)</code></b>
Calculates happiness for girls of type Normal
Overrides: Girls.Girl.happinessCalculator

## 17 Module q9

### 17.1 Functions

<b>main()</b>
main function for question 9 , creates an object of class 'chooseFromBest' prints the formed couples , the changes are made in the log file 'logGifting.txt' durin the program

## 18 Module reader

### 18.1 Variables

Name	Description
arrBoys	Value: []
arrGirls	Value: []
Gifts	Value: []
--package--	Value: None

### 18.2 Class read

#### 18.2.1 Methods

```
--init--(self)
```

```
readcsvfile(self, charType)
```

## **19   Script script-girls\_csv**

## 20 Script script-logGifting.txt

### 20.1 Variables

Name	Description
--package--	<b>Value:</b> None

## Index

boyGeek (*module*), 5  
    boyGeek.boyGeek (*class*), 5  
boyGenerous (*module*), 6  
    boyGenerous.boyGenerous (*class*), 6  
boyMiser (*module*), 7  
    boyMiser.boyMiser (*class*), 7  
Boys (*module*), 2  
    Boys.Boy (*class*), 2  
        Boys.Boy.\_\_init\_\_ (*method*), 2  
        Boys.Boy.gifting (*method*), 2  
        Boys.Boy.happinessCalculator (*method*), 2  
chooseFromBest (*module*), 8  
    chooseFromBest.formCouples (*class*), 8  
        chooseFromBest.formCouples.kBestItems (*method*), 8  
        chooseFromBest.formCouples.makeCouples (*method*), 8  
couple (*module*), 9  
    couple.couple (*class*), 9  
        couple.couple.\_\_init\_\_ (*method*), 9  
        couple.couple.compatibilityCalculator (*method*), 9  
        couple.couple.happinessCalculator (*method*), 9  
driver (*module*), 10  
    driver.generateRandomInput (*function*), 10  
generator (*module*), 11  
    generator.randomGenerator (*class*), 11  
        generator.randomGenerator.\_\_init\_\_ (*method*), 11  
        generator.randomGenerator.generateBoys (*method*), 11  
        generator.randomGenerator.generateGifts (*method*), 11  
        generator.randomGenerator.generateGirls (*method*), 11  
giftEssential (*module*), 12  
    giftEssential.giftEssential (*class*), 12  
giftLuxury (*module*), 13  
    giftLuxury.giftLuxury (*class*), 13  
Gifts (*module*), 3  
    Gifts.Gift (*class*), 3  
        Gifts.Gift.\_\_init\_\_ (*method*), 3  
giftUtility (*module*), 14  
    giftUtility.giftUtility (*class*), 14  
girlChoosy (*module*), 15  
    girlChoosy.girlChoosy (*class*), 15  
girlDesperate (*module*), 16  
    girlDesperate.girlDesperate (*class*), 16  
girlNormal (*module*), 17  
    girlNormal.girlNormal (*class*), 17  
Girls (*module*), 4  
    Girls.Girl (*class*), 4  
        Girls.Girl.\_\_init\_\_ (*method*), 4  
        Girls.Girl.happinessCalculator (*method*), 4  
q9 (*module*), 18  
    q9.main (*function*), 18  
reader (*module*), 19  
    reader.read (*class*), 19  
        reader.read.\_\_init\_\_ (*method*), 19  
        reader.read.readcsvfile (*method*), 19  
script-girls\_csv (*script*), 20  
script-logGifting\_txt (*script*), 21