

# PPL Assignment - Question8

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 cMaker</b>	<b>7</b>
6.1 Variables . . . . .	7
6.2 Class coupleMaker . . . . .	7
6.2.1 Methods . . . . .	7
<b>7 Module couple</b>	<b>8</b>
7.1 Variables . . . . .	8
7.2 Class couple . . . . .	8
7.2.1 Methods . . . . .	8
<b>8 Module driver</b>	<b>9</b>

8.1	Functions . . . . .	9
8.2	Variables . . . . .	9
<b>9</b>	<b>Module generator</b>	<b>10</b>
9.1	Variables . . . . .	10
9.2	Class randomGenerator . . . . .	10
9.2.1	Methods . . . . .	10
<b>10</b>	<b>Module giftEssential</b>	<b>11</b>
10.1	Variables . . . . .	11
10.2	Class giftEssential . . . . .	11
10.2.1	Methods . . . . .	11
<b>11</b>	<b>Module giftLuxury</b>	<b>12</b>
11.1	Variables . . . . .	12
11.2	Class giftLuxury . . . . .	12
11.2.1	Methods . . . . .	12
<b>12</b>	<b>Module giftUtility</b>	<b>13</b>
12.1	Variables . . . . .	13
12.2	Class giftUtility . . . . .	13
12.2.1	Methods . . . . .	13
<b>13</b>	<b>Module girlChoosy</b>	<b>14</b>
13.1	Variables . . . . .	14
13.2	Class girlChoosy . . . . .	14
13.2.1	Methods . . . . .	14
<b>14</b>	<b>Module girlDesperate</b>	<b>15</b>
14.1	Variables . . . . .	15
14.2	Class girlDesperate . . . . .	15
14.2.1	Methods . . . . .	15
<b>15</b>	<b>Module girlNormal</b>	<b>16</b>
15.1	Variables . . . . .	16
15.2	Class girlNormal . . . . .	16
15.2.1	Methods . . . . .	16
<b>16</b>	<b>Module q8</b>	<b>17</b>
16.1	Functions . . . . .	17
<b>17</b>	<b>Module reader</b>	<b>18</b>
17.1	Variables . . . . .	18
17.2	Class read . . . . .	18
17.2.1	Methods . . . . .	18
<b>18</b>	<b>Script script-boys_csv</b>	<b>19</b>
<b>19</b>	<b>Script script-girls_csv</b>	<b>20</b>
<b>20</b>	<b>Script script-logGifting_txt</b>	<b>21</b>

# 1 Module Boys

## 1.1 Variables

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

## 1.2 Class Boy

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

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

<b><code>__init__(self, boy)</code></b>
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__
<b><code>happinessCalculator(self, gIntelligence)</code></b>
Calculates happiness for Geek boys Overrides: Boys.Boy.happinessCalculator
<b><code>gifting(self, gMaintenanceCost, Gifts, giftBasket)</code></b>
Sets up the gift basket for Miser boys Overrides: Boys.Boy.gifting
<b><code>NewGifting(self, gMaintenanceCost, Gifts, giftBasket)</code></b>
Allocates gifts with the new gifting strategy
<b><code>getGift(self, type, Gifts, giftBasket)</code></b>
Allocates one gift of each type

## 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

<b><code>--init--(self, boy)</code></b>
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. <code>--init--</code>

<b><code>happinessCalculator(self, gHappiness)</code></b>
Calculates happiness for Generous boys
Overrides: Boys.Boy. <code>happinessCalculator</code>

<b><code>gifting(self, gMaintenanceCost, Gifts, giftBasket)</code></b>
Sets up the gift basket for Generous boys
Overrides: Boys.Boy. <code>gifting</code>

<b><code>NewGifting(self, gMaintenanceCost, Gifts, giftBasket)</code></b>
Allocates gifts with the new gifting strategy

<b><code>getGift(self, type, Gifts, giftBasket)</code></b>
Allocates one gift of each type

## 6 Module cMaker

### 6.1 Variables

Name	Description
Couples	Value: []
brokenUpGirls	Value: []

### 6.2 Class coupleMaker

#### 6.2.1 Methods

**makeCouples(*self*)**

Makes Couples and prints the gifting details of each , here the gifting is done by the new strategy of giving a gift of each type irrespective of the Budget. The log file 'logGifting.txt' is updated during the program



## 7 Module couple

### 7.1 Variables

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

### 7.2 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

#### 7.2.1 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

## 8 Module driver

### 8.1 Functions

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

### 8.2 Variables

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

## 9 Module generator

### 9.1 Variables

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

### 9.2 Class randomGenerator

#### 9.2.1 Methods

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

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

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

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

## 10 Module giftEssential

### 10.1 Variables

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

### 10.2 Class giftEssential

Gifts.Gift —  
                   **giftEssential.giftEssential**

Gift Class for gifttype = 'Essential'

#### 10.2.1 Methods

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

## 11 Module **giftLuxury**

### 11.1 Variables

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

### 11.2 Class **giftLuxury**

Gifts.Gift —  
**giftLuxury.giftLuxury**

Gift Class for gifttype = 'Luxury'

#### 11.2.1 Methods

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

## 12 Module *giftUtility*

### 12.1 Variables

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

### 12.2 Class *giftUtility*

Gifts.Gift —  
**giftUtility.giftUtility**

Gift Class for gifttype = 'Utility'

#### 12.2.1 Methods

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

## 13 Module *girlChoosy*

### 13.1 Variables

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

### 13.2 Class *girlChoosy*

Girls.Girl —  
                     *girlChoosy.girlChoosy*

Girl class for `girlType = 'Choosy'`

#### 13.2.1 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>

## 14 Module *girlDesperate*

### 14.1 Variables

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

### 14.2 Class *girlDesperate*

Girls.Girl —  
                   **girlDesperate.girlDesperate**

Girl class for girlType = 'Choosy'

#### 14.2.1 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



## 15 Module *girlNormal*

### 15.1 Variables

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

### 15.2 Class *girlNormal*

Girls.Girl —  
                     *girlNormal.girlNormal*

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 q8

### 16.1 Functions

<code>main()</code>
---------------------

## 17 Module reader

### 17.1 Variables

Name	Description
arrBoys	Value: []
arrGirls	Value: []
Gifts	Value: []

### 17.2 Class read

#### 17.2.1 Methods

```
__init__(self)
```

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

## 18 Script script-boys\_csv

## **19   Script script-girls\_csv**

## 20 Script script-logGifting.txt

## Index

- boyGeek (*module*), 5
  - boyGeek.boyGeek (*class*), 5
  - boyGeek.boyGeek.getGift (*method*), 5
  - boyGeek.boyGeek.NewGifting (*method*), 5
- boyGenerous (*module*), 6
  - boyGenerous.boyGenerous (*class*), 6
  - boyGenerous.boyGenerous.getGift (*method*), 6
  - boyGenerous.boyGenerous.NewGifting (*method*), 6
- Boys (*module*), 2
  - Boys.Boy (*class*), 2
  - Boys.Boy.\_\_init\_\_ (*method*), 2
  - Boys.Boy.gifting (*method*), 2
  - Boys.Boy.happinessCalculator (*method*), 2
- cMaker (*module*), 7
  - cMaker.coupleMaker (*class*), 7
  - cMaker.coupleMaker.makeCouples (*method*), 7
- couple (*module*), 8
  - couple.couple (*class*), 8
  - couple.couple.\_\_init\_\_ (*method*), 8
  - couple.couple.compatibilityCalculator (*method*), 8
  - couple.couple.happinessCalcuator (*method*), 8
- driver (*module*), 9
  - driver.generateRandomInput (*function*), 9
- generator (*module*), 10
  - generator.randomGenerator (*class*), 10
  - generator.randomGenerator.\_\_init\_\_ (*method*), 10
  - generator.randomGenerator.generateBoys (*method*), 10
  - generator.randomGenerator.generateGifts (*method*), 10
  - generator.randomGenerator.generateGirls (*method*), 10
- giftEssential (*module*), 11
  - giftEssential.giftEssential (*class*), 11
- giftLuxury (*module*), 12
  - giftLuxury.giftLuxury (*class*), 12
- Gifts (*module*), 3
  - Gifts.Gift (*class*), 3
  - Gifts.Gift.\_\_init\_\_ (*method*), 3
- giftUtility (*module*), 13
  - giftUtility.giftUtility (*class*), 13
- girlChoosy (*module*), 14
  - girlChoosy.girlChoosy (*class*), 14
- girlDesperate (*module*), 15
  - girlDesperate.girlDesperate (*class*), 15
- girlNormal (*module*), 16
  - girlNormal.girlNormal (*class*), 16
- Girls (*module*), 4
  - Girls.Girl (*class*), 4
  - Girls.Girl.\_\_init\_\_ (*method*), 4
  - Girls.Girl.happinessCalculator (*method*), 4
- q8 (*module*), 17
  - q8.main (*function*), 17
- reader (*module*), 18
  - reader.read (*class*), 18
  - reader.read.\_\_init\_\_ (*method*), 18
  - reader.read.readcsvfile (*method*), 18
- script-boys\_csv (*script*), 19
- script-girls\_csv (*script*), 20
- script-logGifting\_txt (*script*), 21