

# PPL Assignment - Question11

## API Documentation

April 19, 2017

## Contents

<b>Contents</b>	<b>1</b>
<b>1 Package error-2-logicalError</b>	<b>2</b>
1.1 Modules . . . . .	2
1.2 Variables . . . . .	2
<b>2 Module error-2-logicalError.boy'</b>	<b>3</b>
2.1 Variables . . . . .	3
2.2 Class Boy . . . . .	3
2.2.1 Methods . . . . .	3
2.3 Class Geek . . . . .	3
2.3.1 Methods . . . . .	3
2.4 Class Generous . . . . .	4
2.4.1 Methods . . . . .	4
2.5 Class Miser . . . . .	4
2.5.1 Methods . . . . .	4
<b>3 Module error-2-logicalError.couple'</b>	<b>5</b>
3.1 Variables . . . . .	5
3.2 Class Boy . . . . .	5
3.2.1 Methods . . . . .	5
3.3 Class Couple . . . . .	5
3.3.1 Methods . . . . .	5
3.4 Class Girl . . . . .	5
3.4.1 Methods . . . . .	6
3.4.2 Instance Variables . . . . .	6
<b>4 Module error-2-logicalError.gift'</b>	<b>7</b>
4.1 Variables . . . . .	7
4.2 Class Gift . . . . .	7
4.2.1 Methods . . . . .	7
4.3 Class essentialGift . . . . .	7
4.3.1 Methods . . . . .	7
4.4 Class luxuryGift . . . . .	7
4.4.1 Methods . . . . .	8
4.5 Class utilityGift . . . . .	8
4.5.1 Methods . . . . .	8

<b>5</b>	<b>Module error-2-logicalError.girl</b>	<b>9</b>
5.1	Variables . . . . .	9
5.2	Class Girl . . . . .	9
5.2.1	Methods . . . . .	9
5.2.2	Instance Variables . . . . .	9
5.3	Class Choosy . . . . .	9
5.3.1	Methods . . . . .	10
5.3.2	Instance Variables . . . . .	10
5.4	Class Normal . . . . .	10
5.4.1	Methods . . . . .	10
5.4.2	Instance Variables . . . . .	10
5.5	Class Desperate . . . . .	11
5.5.1	Methods . . . . .	11
5.5.2	Instance Variables . . . . .	11
<b>6</b>	<b>Module error-2-logicalError.q11'</b>	<b>12</b>
6.1	Functions . . . . .	12
6.2	Variables . . . . .	12
<b>7</b>	<b>Module error-2-logicalError.utility</b>	<b>13</b>
7.1	Functions . . . . .	13
7.2	Variables . . . . .	13
<b>8</b>	<b>Script script-boys_list.txt</b>	<b>14</b>
<b>9</b>	<b>Script script-couple_list.txt</b>	<b>15</b>
<b>10</b>	<b>Script script-gift_list.txt</b>	<b>16</b>
<b>11</b>	<b>Script script-girls_list.txt</b>	<b>17</b>

# 1 Package error-2-logicalError

## 1.1 Modules

- **boy** (Section ??, p. ??)
- **boy'** (Section 2, p. 3)
- **couple** (Section ??, p. ??)
- **couple'** (Section 3, p. 5)
- **gift** (Section ??, p. ??)
- **gift'** (Section 4, p. 7)
- **girl** (Section 5, p. 9)
- **q11** (Section ??, p. ??)
- **q11'** (Section 6, p. 12)
- **utility** (Section 7, p. 13)

## 1.2 Variables

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

## 2 Module error-2-logicalError.boy'

### 2.1 Variables

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

### 2.2 Class Boy

**Known Subclasses:** error-2-logicalError.boy.Geek, error-2-logicalError.boy.Generous, error-2-logicalError.boy.Miser

:param boy: a boy object :param name: name attribute of the boy object :param attractiveness: attractiveness of the boy :param budget: Budget the boy has :param intelligence\_level: Intelligence level of the boy :param min\_attraction\_requirement: Attraction Requirement of the boy :param committed: Commitment status of the boy 0-single 1-committed :param happiness: Happiness of the boy

#### 2.2.1 Methods

```
__init__(self, name, attractiveness, budget, intelligence_level, min_attraction_requirement,
committed, happiness)
```

```
commit(self)
```

```
isCommitted(self)
```

### 2.3 Class Geek

```

error-2-logicalError.boy.Boy └─
                               error-2-logicalError.boy.Geek

```

#### 2.3.1 Methods

```
__init__(self, name, attractiveness, budget, intelligence_level, min_attraction_requirement,
committed, happiness)
```

Overrides: error-2-logicalError.boy.Boy.\_\_init\_\_

```
getHappiness(self, girl, total_gifts)
```

*Inherited from error-2-logicalError.boy.Boy(Section 3.2)*

commit(), isCommitted()

## 2.4 Class Generous



### 2.4.1 Methods

<b><code>__init__</code></b> ( <i>self</i> , <i>name</i> , <i>attractiveness</i> , <i>budget</i> , <i>intelligence_level</i> , <i>min_attraction_requirement</i> , <i>committed</i> , <i>happiness</i> ) Overrides: error-2-logicalError.boy.Boy.__init__
---

<b><code>getHappiness</code></b> ( <i>self</i> , <i>girl</i> , <i>total_gifts</i> )
---

*Inherited from error-2-logicalError.boy.Boy(Section 3.2)*

`commit()`, `isCommitted()`

## 2.5 Class Miser



### 2.5.1 Methods

<b><code>__init__</code></b> ( <i>self</i> , <i>name</i> , <i>attractiveness</i> , <i>budget</i> , <i>intelligence_level</i> , <i>min_attraction_requirement</i> , <i>committed</i> , <i>happiness</i> ) Overrides: error-2-logicalError.boy.Boy.__init__
---

<b><code>getHappiness</code></b> ( <i>self</i> , <i>girl</i> , <i>total_gifts</i> )
---

*Inherited from error-2-logicalError.boy.Boy(Section 3.2)*

`commit()`, `isCommitted()`

### 3 Module error-2-logicalError.couple'

#### 3.1 Variables

Name	Description
<code>--package--</code>	<b>Value:</b> 'error-2-logicalError'

#### 3.2 Class Boy

**Known Subclasses:** error-2-logicalError.boy.Geek, error-2-logicalError.boy.Generous, error-2-logicalError.boy.Miser

:param boy: a boy object :param name: name attribute of the boy object :param attractiveness: attractiveness of the boy :param budget: Budget the boy has :param intelligence\_level: Intelligence level of the boy :param min\_attraction\_requirement: Attraction Requirement of the boy :param committed: Commitment status of the boy 0-single 1-committed :param happiness: Happiness of the boy

##### 3.2.1 Methods

```
--init--(self, name, attractiveness, budget, intelligence_level,
min_attraction_requirement, committed, happiness)
```

```
commit(self)
```

```
isCommitted(self)
```

#### 3.3 Class Couple

##### 3.3.1 Methods

```
--init--(self, boy, girl, happiness, compatibility)
```

#### 3.4 Class Girl

**Known Subclasses:** error-2-logicalError.girl.Choosy, error-2-logicalError.girl.Desperate, error-2-logicalError.girl.Normal

:param name: name attribute of the Girl object :param attractiveness: attractiveness of the Girl :param maintainance\_budget: maintainance Budget the girl requires :param intelligence\_level: Intelligence level of the Girl :param min\_attraction\_requirement: Attraction Requirement of the girl :param committed: Commitment status of the Girl 0-single 1-committed :param happiness: Happiness of the Girl

### 3.4.1 Methods

```
__init__(self, name, attractiveness, maintainance_budget, intelligence_level,  
criterion, committed, happiness)
```

```
isCommitted(self)
```

```
commit(self)
```

### 3.4.2 Instance Variables

Name	Description
happiness	Method to find commitment status
committed	Subclass of Girl class for Choosy

## 4 Module error-2-logicalError.gift'

### 4.1 Variables

Name	Description
<code>__package__</code>	<b>Value:</b> None

### 4.2 Class Gift

**Known Subclasses:** error-2-logicalError.gift.essentialGift, error-2-logicalError.gift.luxuryGift, error-2-logicalError.gift.utilityGift

:param: price : Price of the gift :param: value: Value of the gift

#### 4.2.1 Methods

<code>__init__(self, price, value, used=False)</code>
---

### 4.3 Class essentialGift

```

error-2-logicalError.gift.Gift └─
                                error-2-logicalError.gift.essentialGift

```

#### 4.3.1 Methods

<code>__init__(self, price, value, used, Type)</code> Overrides: error-2-logicalError.gift.Gift.__init__
---

### 4.4 Class luxuryGift

```

error-2-logicalError.gift.Gift └─
                                error-2-logicalError.gift.luxuryGift

```



#### 4.4.1 Methods

`--init--(self, price, value, used, difficulty, rating, Type)`

Overrides: `error-2-logicalError.gift.Gift.__init__`

### 4.5 Class *utilityGift*

`error-2-logicalError.gift.Gift` └─ **`error-2-logicalError.gift.utilityGift`**

#### 4.5.1 Methods

`--init--(self, price, value, used, utility_value, utility_class, Type)`

Overrides: `error-2-logicalError.gift.Gift.__init__`

## 5 Module error-2-logicalError.girl

### 5.1 Variables

Name	Description
<code>--package--</code>	Value: 'error-2-logicalError'

### 5.2 Class Girl

**Known Subclasses:** error-2-logicalError.girl.Choosy, error-2-logicalError.girl.Desperate, error-2-logicalError.girl.Normal

:param name: name attribute of the Girl object :param attractiveness: attractiveness of the Girl :param maintainance\_budget: maintainance Budget the girl requires :param intelligence\_level: Intelligence level of the Girl :param min\_attraction\_requirement: Attraction Requirement of the girl :param committed: Commitment status of the Girl 0-single 1-committed :param happiness: Happiness of the Girl

#### 5.2.1 Methods

```
--init--(self, name, attractiveness, maintainance_budget, intelligence_level,
criterion, committed, happiness)
```

```
isCommitted(self)
```

```
commit(self)
```

#### 5.2.2 Instance Variables

Name	Description
happiness	Method to find commitment status
committed	Subclass of Girl class for Choosy

### 5.3 Class Choosy

```
error-2-logicalError.girl.Girl └─ error-2-logicalError.girl.Choosy
```

### 5.3.1 Methods

```
__init__(self, name, attractiveness, maintainance_budget, intelligence_level,
criterion, committed, happiness)
```

Overrides: error-2-logicalError.girl.Girl.\_\_init\_\_

```
getHappiness(self, boy, total_gifts, total_gift_value)
```

*Inherited from error-2-logicalError.girl.Girl (Section 5.2)*

commit(), isCommitted()

### 5.3.2 Instance Variables

Name	Description
<i>Inherited from error-2-logicalError.girl.Girl (Section 5.2)</i>	
committed, happiness	

## 5.4 Class Normal

```
error-2-logicalError.girl.Girl └─ error-2-logicalError.girl.Normal
```

### 5.4.1 Methods

```
__init__(self, name, attractiveness, maintainance_budget, intelligence_level,
criterion, committed, happiness)
```

Overrides: error-2-logicalError.girl.Girl.\_\_init\_\_

```
getHappiness(self, boy, total_gifts, total_gift_value)
```

*Inherited from error-2-logicalError.girl.Girl (Section 5.2)*

commit(), isCommitted()

### 5.4.2 Instance Variables

Name	Description
<i>Inherited from error-2-logicalError.girl.Girl (Section 5.2)</i>	

*continued on next page*

Name	Description
committed, happiness	

## 5.5 Class Desperate

error-2-logicalError.girl.Girl —  
**error-2-logicalError.girl.Desperate**

### 5.5.1 Methods

**\_\_init\_\_**(*self, name, attractiveness, maintainance\_budget, intelligence\_level, criterion, committed, happiness*)

Overrides: error-2-logicalError.girl.Girl.\_\_init\_\_

**getHappiness**(*self, boy, total\_gifts, total\_gift\_value*)

*Inherited from error-2-logicalError.girl.Girl (Section 5.2)*

commit(), isCommitted()

### 5.5.2 Instance Variables

Name	Description
<i>Inherited from error-2-logicalError.girl.Girl (Section 5.2)</i>	
committed, happiness	

## 6 Module error-2-logicalError.q11'

### 6.1 Functions

<code>generateBoyList()</code>
--------------------------------

<code>generateGirlList()</code>
---------------------------------

<code>giveGifts(<i>boy_list</i>, <i>girl_list</i>)</code>
---

<code>q4()</code>
-------------------

<code>valentine(<i>k</i>, <i>couple_list</i>, <i>boy_list</i>, <i>girl_list</i>)</code>
---

### 6.2 Variables

Name	Description
<code>--package--</code>	<b>Value:</b> 'error-2-logicalError'

## 7 Module error-2-logicalError.utility

### 7.1 Functions

```
randomName(size=6, chars= 'ABCDEFGHIJKLMNOPQRSTUVWXYZ')
```

```
generateBoyList()
```

```
generateGirlList()
```

```
generateGiftList()
```

```
makeCouples(mode, boy_list, girl_list)
```

```
giveGifts(boy_list, girl_list)
```

```
k_most_happy_couples(couple_list)
```

```
k_least_happy_couples(couple_list)
```

```
valentine(k, couple_list, boy_list, girl_list)
```

### 7.2 Variables

Name	Description
--package--	Value: 'error-2-logicalError'

## 8 Script script-boys\_list.txt

## 9 Script script-couple\_list.txt



## 10 Script script-gift\_list.txt

## **11 Script script-girls\_list.txt**

## Index

- error-2-logicalError (*package*), 2
  - error-2-logicalError.boy (*module*)
    - error-2-logicalError.boy.Boy (*class*), 3, 5
    - error-2-logicalError.boy.Geek (*class*), 3
    - error-2-logicalError.boy.Generous (*class*), 3–4
    - error-2-logicalError.boy.Miser (*class*), 4
  - error-2-logicalError.boy' (*module*), 3–4
  - error-2-logicalError.couple (*module*)
    - error-2-logicalError.couple.Couple (*class*), 5
  - error-2-logicalError.couple' (*module*), 5–6
  - error-2-logicalError.gift (*module*)
    - error-2-logicalError.gift.essentialGift (*class*), 7
    - error-2-logicalError.gift.Gift (*class*), 7
    - error-2-logicalError.gift.luxuryGift (*class*), 7–8
    - error-2-logicalError.gift.utilityGift (*class*), 8
  - error-2-logicalError.gift' (*module*), 7–8
  - error-2-logicalError.girl (*module*), 9–11
    - error-2-logicalError.girl.Choosy (*class*), 9–10
    - error-2-logicalError.girl.Desperate (*class*), 11
    - error-2-logicalError.girl.Girl (*class*), 5–6, 9
    - error-2-logicalError.girl.Normal (*class*), 10–11
  - error-2-logicalError.q11 (*module*)
    - error-2-logicalError.q11.q4 (*function*), 12
  - error-2-logicalError.q11' (*module*), 12
  - error-2-logicalError.utility (*module*), 13
    - error-2-logicalError.utility.generateBoyList (*function*), 12, 13
    - error-2-logicalError.utility.generateGiftList (*function*), 13
    - error-2-logicalError.utility.generateGirlList (*function*), 12, 13
    - error-2-logicalError.utility.giveGifts (*function*), 12, 13
    - error-2-logicalError.utility.k\_least\_happy\_couples (*function*), 13
    - error-2-logicalError.utility.k\_most\_happy\_couples (*function*), 13
    - error-2-logicalError.utility.makeCouples (*function*), 13
    - error-2-logicalError.utility.randomName (*function*), 13
    - error-2-logicalError.utility.valentine (*function*), 12, 13
- script-boys\_list.txt (*script*), 14
- script-couple\_list.txt (*script*), 15
- script-gift\_list.txt (*script*), 16
- script-girls\_list.txt (*script*), 17