**Test case**

Media Class and Magazine Class

1. Media() :

This is the method that automatically established by the system, so we don’t have to test it here and we assume it’s good.

2. Through instantiating a Magazine object to test the accessor methods ,Media(String title , String author, String pubDate, double price, LocalDate borrowDate) and Magazine (String title , String author, String pubDate, double price, LocalDate borrowDate) ,

Because the accessor methods has been put into the Media(title…) to give the attributes default value.

1. Analyse each rule/constraint for the valid and invalid cases. Define and number the equivalency classes within the valid and invalid cases

|  |  |  |
| --- | --- | --- |
| **Rules/Constraints** | **Valid Equivalence Classes** | **Invalid Equivalence Classes** |
| Title ,author , pubDate are string | 1: type is string | 2: null  3 :empty  4 : other type except string |
| borrowDate is LocalDate | 5:type is LocalDate | 6: other type except LocalDate |
| borrowDate can not late than today | 7:all the date until today | 8 : later than toady |
| Price is double | 9:type is double | 10:other type except double |
| Price cannot less than 0.0 | 11: price >0.0 | 12: price = 0.0  13: price < 0.0 |

2 . Determine test data that maps back to each of your equivalency classes.

|  |  |
| --- | --- |
| Test value | Test equivalence |
| title = “Bicycling, March 2020” ;  author = "Richard"; pubDate =”2020,3,1” ; price= 6.76 ;  borrowDate = 2020,3.29 | 1,5,7,9,11 |
| title ,author , pubDate = null (any one of them ) | 2 |
| title ,author , pubDate = “” (any one of them ) | 3 |
| title ,author , pubDate = 10(any one of them ) | 4 |
| borrowDate = “2020,9,3” | 6 |
| borrowDate = LocalDate.of(2021 , 3, 29) | 8 |
| price = “6.79” | 10 |
| price = 0.0 | 12 |
| Price = -6.79 | 13 |

3. Capture your test cases

|  |  |  |  |
| --- | --- | --- | --- |
| **Operation** | **Purpose** | **Object State** | **Expected Result** |
| Magazine m = new Magazine("Bicycling, March 2020", "Richard",“2020,3,1”,6.76,LocalDate.of(2020, 3. 29)); | To create a magazine object | title = “Bicycling, March 2020” ;  author = "Richard"; pubDate =”2020,3,1” ; price= 6.76 ;  borrowDate = 2020,3.29 | Create a Magazine object |
| Magazine m = new Magazine(null, “2020,3,1”,6.76,LocalDate.of(2020, 3. 29)); | Title ,author or pubDate is null |  | throw Illegal Argument Exception |
| Magazine m = new Magazine(("Bicycling, March 2020", "Richard",“ ” ,6.76,LocalDate.of(2020, 3. 29)); | Title ,author or pubDate is empty |  | throw Illegal Argument Exception |
| Magazine m = new Magazine("Bicycling, March 2020", 10 ,“2020,3,1”,6.76,LocalDate.of(2020, 3. 29)); | Title ,author or pubDate is other type except string |  | throw Illegal Argument Exception |
| Magazine m = new Magazine("Bicycling, March 2020", "Richard",“2020,3,1”,6.76,”2020, 3. 29”); | borrowDate is other type except LocalDate |  | throw Illegal Argument Exception |
| Magazine m = new Magazine("Bicycling, March 2020", "Richard",“2020,3,1”,6.76,LocalDate.of(2021, 3. 29)); | borrowDate is later than toady |  | throw Illegal Argument Exception |
| Magazine m = new Magazine("Bicycling, March 2020", "Richard",“2020,3,1”,6.76,LocalDate.of(2021, 3. 29)); |  |  |  |
| Magazine m = new Magazine("Bicycling, March 2020", "Richard",“2020,3,1”,”6.76”,LocalDate.of(2020, 3. 29)); | price is other type except double |  | throw Illegal Argument Exception |
| Magazine m = new Magazine("Bicycling, March 2020", "Richard",“2020,3,1”,0.0,LocalDate.of(2020, 3. 29)); | price=0.0 |  | throw Illegal Argument Exception |
| Magazine m = new Magazine("Bicycling, March 2020", "Richard",“2020,3,1”,  -6.76,LocalDate.of(2020, 3. 29)); | Price<0.0 |  | throw Illegal Argument Exception |
| Magazine m = new Magazine("Bicycling, March 2020", "Richard",“2020,3,1”,6.76,LocalDate.of(2020, 3. 29)); | To create a magazine object | title = “Bicycling, March 2020” ;  author = "Richard"; pubDate =”2020,3,1” ; price= 6.76 ;  borrowDate = 2020,3.29 | Create a Magazine object |
| m.getTitle() | To verify instantiation and accessor method |  | "Bicycling, March 2020" |
| m.getAuthor() | To verify instantiation and accessor method |  | "Richard" |
| m.getPubDate() | To verify instantiation and accessor method |  | “2020, 3, 1” |
| media.getPrice() | To verify instantiation and accessor method |  | 6.76 |
| media.getBorrowDate() | To verify instantiation and accessor method |  | 2020,3.29 |

3. getMediaType():String and setMediaType():void

1. Analyse each rule/constraint for the valid and invalid cases. Define and number the equivalency classes within the valid and invalid cases

|  |  |  |
| --- | --- | --- |
| **Rules/Constraints** | **Valid Equivalence Classes** | **Invalid Equivalence Classes** |
| The value of mediaType is  valid | 1:The value of mediaType is  “M” ,  2:“P’’  3:”E”  or 4: “G”  (case-sensitive) | 5: other string except “M” , “P’’ or ’E’ or’ G’(case-sensitive) |

1. Determine test data that maps back to each of your equivalency classes.

|  |  |
| --- | --- |
| Test value | Test equivalence |
| mediaType = “M” or “m” | 1 |
| mediaType = “P” or “ p” | 2 |
| mediaType = “E” or “e” | 3 |
| mediaType = “G” or “g” | 4 |
| mediaType = “aa” | 5 |

Table 1

1. Capture your test cases

|  |  |  |  |
| --- | --- | --- | --- |
| **Operation** | **Purpose** | **Object State** | **Expected Result** |
| PaperBook p = new PaperBook () | To Create a PaperBook object | All the attribute is default |  |
| p. setType(‘P’)or m. setType(‘p’) | Test type =’P’ or Test type =’p’ |  |  |
| p.getType()=”Paperback” |  |  | mediaType = “Paperback” |
| Magazine m = new Magazine() | To Create a magazine object |  |  |
| media. setType(‘M’)or media. setType(‘m’) | Test type =’M’ or Test type =’m’ |  |  |
| m.getType()=”Magazine” |  |  | mediaType = “Magazine” |
| Ebook e =new Ebook() | To Create a Ebook  object |  |  |
| e. setType(‘E’)or e. setType(‘e’) | Test type =’E’ or Test type =’e’ |  |  |
| e.getType()=”Ebook” |  |  | mediaType = ”Ebook” |
| Game g =new Game() | To Create a Game  object |  |  |
| g. setType(‘G’)or g. setType(‘g’) | Test type =’G’ or Test type =’g’ |  |  |
| g.getType()=”Game” |  |  | mediaType = ” Game” |
| g. setType() | Test type =”aa” |  | throw Illegal Argument Exception |

4. Media(String mediaType, String title , Stirng author,String pubDate, double price, LocalDate borrowDate)

Becase Media is abstract class ,it cannot instantiate ,so combine Media(String title ,… LocalDate borrowDate) and method 3 to verify its correctness.

5.borrow()

Analyze each rule/constraint for the valid and invalid cases.

Define equivalency partitions within the valid and invalid cases

|  |  |  |
| --- | --- | --- |
| **Rules/Constraints** | **Valid Equivalence Classes** | **Invalid Equivalence Classes** |
| borrowe date isvalid date | 1 valid date | 2 null |

Determine test data that maps back to each of your equivalency partitions

|  |  |
| --- | --- |
| **Test value** | **Test equivalence # mapping** |
| borrow date is 2020.3.1 | 1 |
| borrow date is null | 2 |

Capture your test cases

|  |  |  |  |
| --- | --- | --- | --- |
| **Operation** | **Purpose** | **Object State** | **Expected Result** |
| Magazine m = new Magazine("Bicycling, March 2020", "Richard",“2020,3,1”,6.76,LocalDate.of(2020, 3. 29)); | To create a Magazine object | title = “Bicycling, March 2020” ;  author = "Richard"; pubDate =”2020,3,1” ; price= 6.76 ;  borrowDate = 2020,3.29 | A new Magazine object |
| m.borrow() | To verify instantiation and borrow method |  | Not Throw Exception |
| Magazine m = new Magazine("Bicycling, March 2020", "Richard",“2020,3,1”,6.76,null) |  |  | Throw Exception |

Other method is same