Chapter 7

Exercise 7.2:
UML Class Diagram for Exercise 7.2

<table>
<thead>
<tr>
<th>Fan</th>
<th>Constant.</th>
</tr>
</thead>
<tbody>
<tr>
<td>+SLOW = 1</td>
<td>Constant.</td>
</tr>
<tr>
<td>+MEDIUM = 2</td>
<td>Constant.</td>
</tr>
<tr>
<td>+FAST = 3</td>
<td>The speed of this fan (default 1).</td>
</tr>
<tr>
<td>-speed: int</td>
<td>Indicates whether the fan is on (default false).</td>
</tr>
<tr>
<td>-on: boolean</td>
<td>The radius of this fan (default 5).</td>
</tr>
<tr>
<td>-radius: double</td>
<td>The color of this fan (default white).</td>
</tr>
<tr>
<td>-color: String</td>
<td></td>
</tr>
</tbody>
</table>

Constructs a fan with default values.
Returns the speed of this fan.
Sets a new speed for this fan.
Returns true if this fan is on.
Sets this fan on to true or false.
Returns the radius of this fan.
Sets a new radius for this fan.
Returns the color of this fan.
Sets a new color for this fan.
Returns a string representation for this fan.
## Exercise 7.4:
### UML Class Diagram for Exercise 7.4

<table>
<thead>
<tr>
<th>Stock</th>
<th>Stock symbol: String</th>
<th>The symbol of this stock.</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>-name: String</td>
<td>The name of this stock.</td>
</tr>
<tr>
<td></td>
<td>-previousClosingPrice: double</td>
<td>The previous closing price of this stock.</td>
</tr>
<tr>
<td></td>
<td>-currentPrice: double</td>
<td>The current price of this stock.</td>
</tr>
<tr>
<td></td>
<td>+Stock(symbol: String, name: String)</td>
<td>Constructs a stock with a specified symbol and a name.</td>
</tr>
<tr>
<td></td>
<td>+getSymbol():String</td>
<td>Returns the symbol of this stock.</td>
</tr>
<tr>
<td></td>
<td>+getName():String</td>
<td>Returns the name of this stock.</td>
</tr>
<tr>
<td></td>
<td>+getPreviousClosingPrice(): double</td>
<td>Returns the previous closing price of this stock.</td>
</tr>
<tr>
<td></td>
<td>+getCurrentPrice(): double</td>
<td>Returns the current price of this stock.</td>
</tr>
<tr>
<td></td>
<td>+setPreviousClosingPrice(price: double): void</td>
<td>Sets the previous closing price of this stock.</td>
</tr>
<tr>
<td></td>
<td>+setCurrentPrice(price: double): void</td>
<td>Sets the current price of this stock.</td>
</tr>
<tr>
<td></td>
<td>+changePercent(): double</td>
<td>Returns the percentage of change of this stock.</td>
</tr>
</tbody>
</table>
Exercise 7.14:
UML Class Diagram for Exercise 7.14

<table>
<thead>
<tr>
<th>Tax</th>
</tr>
</thead>
<tbody>
<tr>
<td>-filingStatus: int</td>
</tr>
<tr>
<td>-brackets: int[][]</td>
</tr>
<tr>
<td>-rates: double[]</td>
</tr>
<tr>
<td>-taxableIncome: double</td>
</tr>
<tr>
<td>+Tax()</td>
</tr>
<tr>
<td>+Tax(filingStatus: int, brackets: int[], rates: double[], taxableIncome: double)</td>
</tr>
<tr>
<td>+getFilingStatus(): int</td>
</tr>
<tr>
<td>+setFilingStatus(filingStatus: int): void</td>
</tr>
<tr>
<td>+getBrackets(): int[][]</td>
</tr>
<tr>
<td>+setBrackets(brackets: int[][]): void</td>
</tr>
<tr>
<td>+getRates(): double[]</td>
</tr>
<tr>
<td>+setRates(rates: double[]): void</td>
</tr>
<tr>
<td>+getTaxableIncome(): double</td>
</tr>
<tr>
<td>+setTaxableIncome(taxableIncome: double): void</td>
</tr>
<tr>
<td>+getTax(): double</td>
</tr>
</tbody>
</table>
Chapter 9

Exercise 9.2:
UML Class Diagram for Exercise 9.2

Person

- name: String
- address: String
- phone: String
- email: String

+ Person()
+ Person(name: String, address: String, phone: String, email: String)
+ getName(): String
+ getAddress(): String
+ getPhone(): String
+ getEmail(): String
+ setName(name: String): void
+ setAddress(address: String): void
+ setPhone(phone: String): void
+ setEmail(email: String): void
+ toString(): String

Student

- status: String

+ Student()
+ Student(name: String, address: String, phone: String, email: String)
+ getStatus(): String
+ setStatus(status: String): void
+ toString(): String

Employee

- office: String
- salary: int
- dateHired: java.util.Date

+ Employee()
+ Employee(name: String, address: String, phone: String, email: String)
+ getOffice(): String
+ getSalary(): int
+ getDateHired(): Date
+ setOffice(office: String): void
+ setSalary(salary: int): void
+ setDateHired(dateHired: Date): void
+ toString(): String

Faculty

- officeHour: String
- rank: String

+ Faculty()
+ Faculty(name: String, address: String, phone: String, email: String)
+ getOfficeHour(): String
+ setOfficeHour(officeHour: String): void
+ getRank(): String
+ setRank(rank: String): void
+ toString(): String
Exercise 9.4:
UML Class Diagram for Exercise 9.4

Account

MyStack  SavingsAccount

MyArrayList

MyStack

isEmpty(): boolean
getSize(): int
peek(): Object
pop(): Object
push(o: Object): Object
search(o: Object): int

Returns true if this stack is empty.
Returns the number of elements in this stack.
Returns the top element in this stack.
Returns and removes the top element in this stack.
Adds a new element to the top of this stack.
Returns the position of the first element in the stack from the top that matches the specified element.
Chapter 10

Exercise 10.2:
UML Class Diagram for Exercise 10.2

![UML Class Diagram for Exercise 10.2]
Exercise 10.6:
UML Class Diagram for Exercise 10.6

```
«interface»
Comparable
+compareTo(): int

GeometricObject

Rectangle
```
Chapter 11

Exercise 11.4:
UML Class Diagram for Exercise 10.4

```
Name
-firstName: String
-lastName: String
+Name(firstName: String, mi: char, lastName: String)
+getFirstName(): String
+getMi(): char
+getLastName(): String
+getFullName(): String

Person
-name: Name
-address: Address
+Person()
+Person(name: Name, address: Address)
+getName(): Name
+setName(name: Name): void
+getAddress(): Address
+setAddress(address: Address): void

Comparable

Student
-major: String
+Student(name: Name, major: String)
+getMajor(): String
+setMajor(major: String): void
```

Diagram: [Diagram of UML class diagram]