Solutions for UML Class Diagrams

Chapter 9

Exercise 9.2: Stock

	Stock
-s'	ymbol: str
	ame: str
-p	reviousClosingPrice: float
•	urrentPrice: float
St	ock(symbol: str, name: str)
ge	tChangePercent(): float
ge	tSymbol(): str
ge	tName(): str
ge	tPreviousClosingPrice(): float
se	tPreviousClosingPrice(price: float): None
ge	tCurrentPrice(): float
se	tCurrentPrice(price: float): None

The symbol of this stock.

The name of this stock.

The previous closing price of this stock.

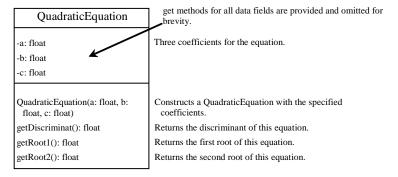
The current price of this stock.

Constructs a stock with a specified symbol and a name.
Returns the percentage of change of this stock.
Returns the symbol of this stock.
Returns the name of this stock.
Returns the previous closing price of this stock.
Sets a new previous closing price of this stock.
Returns the current price of this stock.
Sets a new current price of this stock.

Exercise 9.4: Fan

Fan	
SLOW = 1	Constant defined outside of the class.
$\overline{\text{MEDIUM}} = 2$	Constant defined outside of the class.
FAST = 3	Constant defined outside of the class.
-speed: int	The speed of this fan (default 1).
-on: bool	Indicates whether the fan is on (default false)
-radius: float	The radius of this fan (default 5).
-color: str	The color of this fan (default white).
Fan()	Constructs a fan with default values.
getSpeed(): int	Returns the speed of this fan.
setSpeed(speed: int): None	Sets a new speed for this fan.
isOn(): bool	Returns true if this fan is on.
setOn(on: bool): None	Sets this fan on to true or false.
getRadius(): float	Returns the radius of this fan.
setRadius(radius: float): None	Sets a new radius for this fan.
getColor(): str	Returns the color of this fan.
setColor(color: str): None	Sets a new color for this fan.

Exercice 9.6: QuadraticEquation



Exercise 9.8: StopWatch

StopWatch	get methods for all data fields are provided and omitted for brevity.
-startTime: float -endTime: float	Start time and end time for the watch.
StopWatch()	Constructs a StopWatch with the specified start and end time.
start(): None	Starts the watch.
stop(): None	Stops the watch.
getElapsedTime(): float	Returns the elapsed time.
i	l .

Exercise 9.10: Time

Time	
-hour: int	The hour for the time.
-minute: int	The minute for the time.
-second: int	The second for the time.
Time()	Constructs Time for the current time.
getHour(): int	Returns the clock hour for the time.
getMinute(): int	Returns the minute for the time.
getSecond(): int	Returns the second for the time.
setTime(elapsTime): void	Sets a new time.

Exercise 9.12: Circle2D

Circle2D		
-x: float		
-y: float		
-radius: float		
Circle 2D(res floor) on floor and disconfloor		
Circle2D(x: float, y: float, radius: float)		
getX(): float		
getY(): float		
setX(x: float): void		
setY(y: float): void		
getRadius(): float		
setRadius(radius: float): void		
getPerimeter(): float		
getArea(): float		
contains(x: float, y: float): bool		
contains(circle: Circle2D): bool		
contains(circle: Circle2D): bool		
lt,le,gt,ge,ne,eq		