

# **Advanced Software Design**

***Applied to a Spreadsheet***

# Modeling with Design and Patterns

## A spreadsheet example (I)

- Imagine we are to design a very simple spreadsheet system, example:

	A	B
1	Airfare	\$485
2	Taxi	\$118
3	Rental Car	\$295
4	Hotel	\$432
5	Meals	\$150
6		
7	<b>Total</b>	<b>\$1,480</b>

A1			Airfare
	A	B	C
1	Airfare	\$485	
2	Taxi	\$118	
3	Rental Car	\$295	
4	Hotel	\$432	
5	Meals	\$150	
6			
7	<b>Total</b>	<b>\$1,480</b>	

- You will need to enter numbers, operations, formulas, references to other cells

B1			485
	A	B	
1	Airfare	\$485	
2	Taxi	\$118	
3	Rental Car	\$295	
4	Hotel	\$432	
5	Meals	\$150	
6			
7	<b>Total</b>	<b>\$1,480</b>	

B4			=135*3+27
	A	B	C
1	Airfare	\$485	
2	Taxi	\$118	
3	Rental Car	\$295	
4	Hotel	\$432	
5	Meals	\$150	
6			
7	<b>Total</b>	<b>\$1,480</b>	

B7			=B1+B2+B3+B4+B5
	A	B	C
1	Airfare	\$485	
2	Taxi	\$118	
3	Rental Car	\$295	
4	Hotel	\$432	
5	Meals	\$150	
6			
7	<b>Total</b>	<b>\$1,480</b>	
8			

# Modeling with Design and Patterns

## A spreadsheet example (II)

❑ A more complex example:

	1	2	3	4
1	Airfare:	6885.15		What we pay to the airlines
2	Taxi:	118		
3	Rental Car:	295.85		
4	Hotel:	432		
5	Meals:	150		All meals combined
6				
7	Sub-Total:	7881		
8	Discount:	0.15		
9	Total:	6698.85		
10	Partners:	4		
11	Months:	12		
12	Installments:	139.559375		

# Modeling with Design and Patterns

## A spreadsheet example (II -- Annotated)

- A more complex example:

	1	2	3	4
1	Airfare:	6885.15	$(200*4)/2+5*(18-1)+0.15$	What we pay to the airlines
2	Taxi:	118	118	
3	Rental Car:	295.85	$295.0 + 0.85$	
4	Hotel:	432	$108.0 * 4.0$	
5	Meals:	150		All meals combined
6				
7	Sub-Total:	7881	$[1,2]+[2,2]+[3,2]+[4,2]+[5,2]$	
8	Discount:	0.15		
9	Total:	6698.85	$[7,2] * (1 - [8,2])$	
10	Partners:	4		
11	Months:	12		
12	Installments:	139.559375	$[9,2] / [10,2] / [11,2]$	

# Modeling with Design and Patterns

## Spreadsheet's Sprint 1—application in Java (1)

- ❑ Start with a general Model
- ❑ The following questions will help you get started
  - What are the main features of a spreadsheet?
  - What are the main concepts?
  - What are the main concept relationships?
- ❑ Build a class diagram representing this problem domain
- ❑ Do **NOT** include any UI
  - There is no time to build a GUI
  - We'll build the spreadsheet from the main() method
  - We'll simply print it out to the Console

# Modeling with Design and Patterns

## Spreadsheet's Sprint 1—application in Java (2)

- ❑ Consider which Design Pattern(s) you may use
- ❑ You don't have to commit to any patterns yet, just list them as a possibility
- ❑ The following table may help you decide:

Requirements	Best Pattern	Reasons/ features obtained

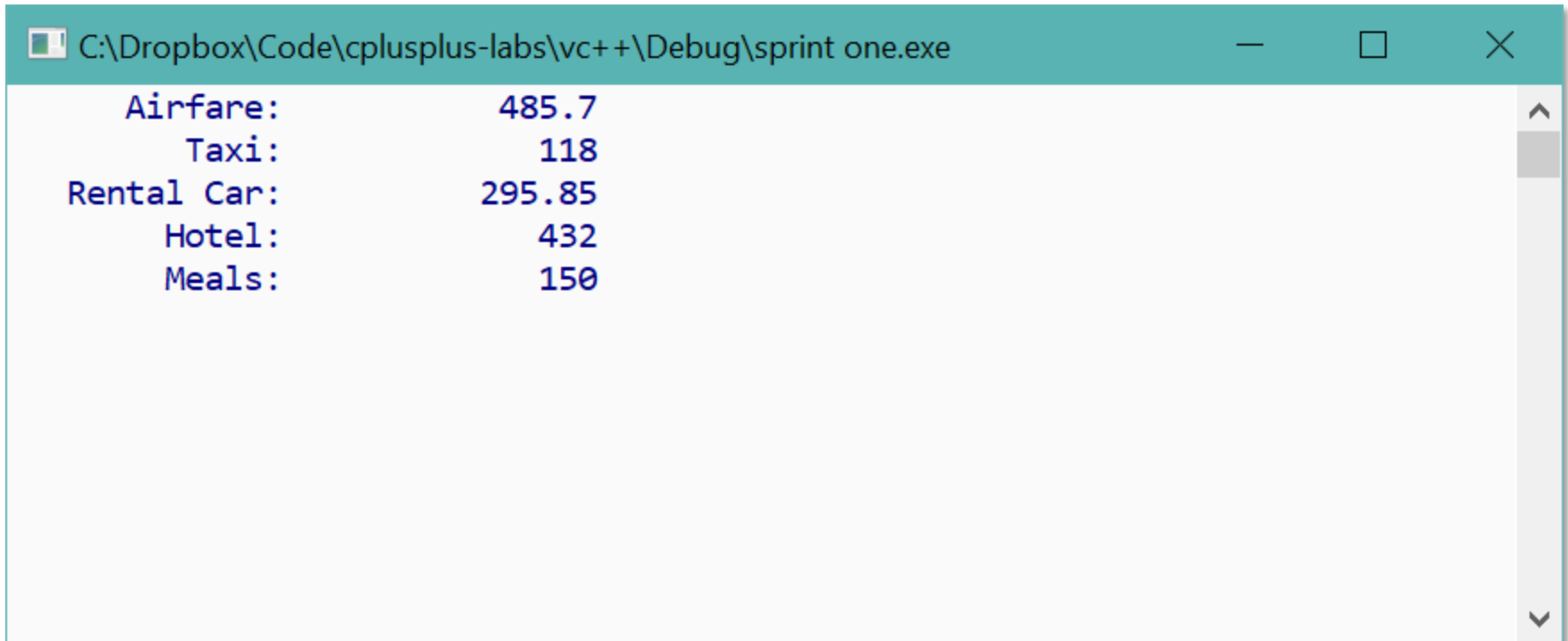
# Modeling with Design and Patterns

## Spreadsheet's Sprint 1—application in Java (3)

- ❑ Build the model for the general solution and organization of your code
- ❑ It should be clear which class(es) your `main()` will invoke, and which methods of those classes
- ❑ Remember your “client code”, `main()` in this case, must be given a very simple task—just a few lines of code
- ❑ Then decide which features will be delivered during each “sprint”
- ❑ Each day will be dedicated to *one or two* sprints
- ❑ Finally, decide upon a small list of features your **Sprint #1** will deliver: next slide provides an idea on how simple it may be. Notice, however, that it must be a “real” spreadsheet—however limited it might be—not just a series of `System.out.println()` statements.
- ❑ Write, test, and submit your diagrams and your code.

# Modeling with Design and Patterns

Spreadsheet's Sprint 1—suggested, typical output



```
C:\Dropbox\Code\cplusplus-labs\vc++\Debug\sprint one.exe

Airfare:      485.7
Taxi:         118
Rental Car:   295.85
Hotel:        432
Meals:        150
```