

CSC 491 / 391

Mobile Application Development for iOS II






Prof. Xiaoping Jia
School of Computing, CDM
DePaul University
xjia@cdm.depaul.edu
[@DePaulSWEng](https://twitter.com/DePaulSWEng)


1

Enum with Associated Values



2

Enum Cases




- Each case of an enum type is defined as a constant

```
enum Status {
    case onTime
    case cancelled
    case delayed
}
```

Delayed by how much?


```
Status.onTime
Status.cancelled
Status.delayed
```



5

5

Enum with Associated Values




- Each case of an enum type may also have one or more *associated values*
 - Provide detailed information about each case
 - Different cases may have different associated values

```
enum Status {
    case onTime
    case cancelled
    case delayed(Int)
}
```

The amount of delay in minutes.


```
Status.onTime
Status.cancelled
Status.delayed(45)
```




6

9

An Example of Associated Value




```
class Flight {
    enum Status {
        case onTime
        case cancelled
        case delayed(Int)
    }
    var code: String
    var status: Status
    init(code: String, status: Status) {
        self.code = code
        self.status = status
    }
}
```



7

10


An Example of Associated Value



```
let flights = [
    Flight(code: "BA 123", status: .onTime),
    Flight(code: "CA 456", status: .cancelled),
    Flight(code: "UA 789", status: .delayed(45)),
]

for f in flights {
    switch f.status {
    case .onTime:
        print("Flight \(f.code) is on time.")
    case .cancelled:
        print("Flight \(f.code) is cancelled.")
    case .delayed:
        print("Flight \(f.code) is delayed.")
    }
}
```

Output:
Flight BA 123 is on time.
Flight CA 456 is cancelled.
Flight UA 789 is delayed.



8

12

An Example of Associated Value

```
for f in flights {  
    switch f.status {  
    case .onTime:  
        print("Flight \(f.code) is on time.")  
    case .cancelled:  
        print("Flight \(f.code) is cancelled.")  
    case .delayed(let min):  
        print("Flight \(f.code) is delayed for \(min) minutes.")  
    }  
}
```

Output:
Flight BA 123 is on time.
Flight CA 456 is cancelled.
Flight UA 789 is delayed for 45 minutes.

DEPAUL UNIVERSITY