

# Cold and Hot Observable




# Observable Creation

```
Observable<Customer> getCustomers() {  
    return Observable.create(new ObservableOnSubscribe<Customer>() {  
  
        public void subscribe(ObservableEmitter<Customer>  
            subscriber) {  
            Connection conn = getConnection();  
            Stmt stmt = conn.createStatement();  
            String sql = "select id, name from customer";  
            ResultSet rs = stmt.executeQuery(sql);  
  
            while (rs.next()) {  
                Long id = rs.getLong("id");  
                String name = rs.getString("name");  
                Customer c = new Customer(id, name);  
                subscriber.onNext(c);  
            }  
  
            subscriber.onComplete();  
            ...  
        }  
    });  
}
```



What  
happens  
when this  
statement is  
executed?

**Observable<Customer> customer = getCustomers();**

# Cold Observable

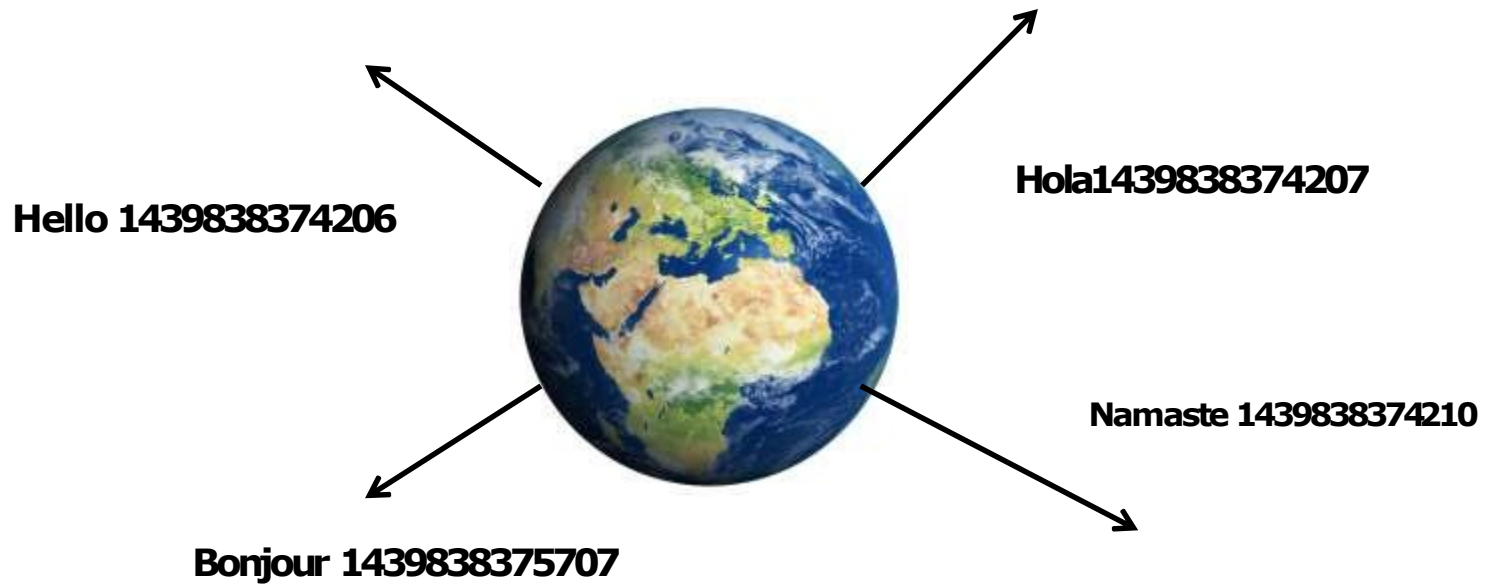
-  **Is the *more common types* of Observables you will deal with**
-  **Does *nothing* until subscribed to**
-  **In example shown, Database was not accessed till a subscriber was present**

# Hot Observable

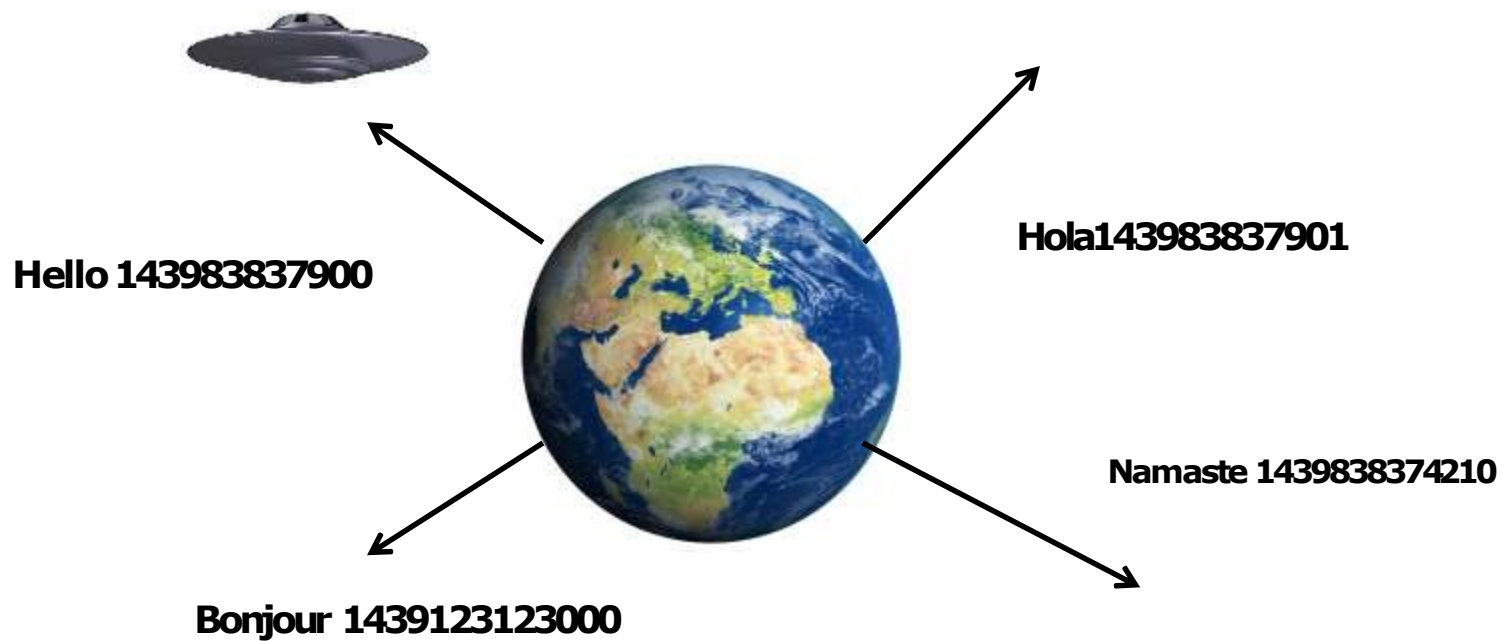
-  ***Hot* observables are "live" sequences that are occurring whether they are being observed or not.**
-  **The canonical example is a mouse, which of course is free to move regardless of whether you subscribe to an *Observable<MouseMoveEvent>* or not.**

## Hot Observable Example – Earth emitting Greetings

### Emitting Greetings in multiple languages +Timestamp





# One Curious Observer



# Second Curious Observer



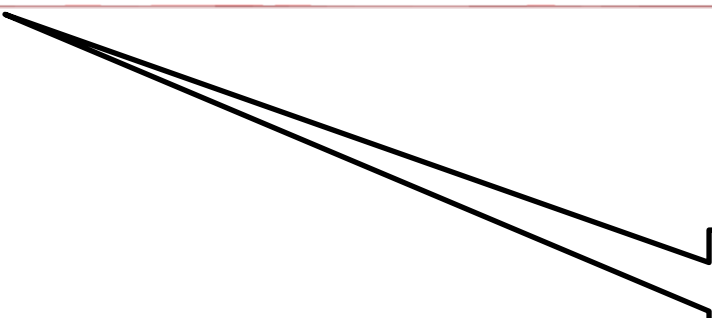
## Hot Observable using - ConnectableObservable

-  **A Connectable Observable resembles an ordinary Observable, except that it does not begin emitting items when it is subscribed to, but only when its *connect()* method is called. Calling *connect()*, makes the Observable 'HOT'.**
-  **ConnectableObservables can be created from many Observable by calling *publish() or replay()***



# Connectable Observable

```
Observable<String> earthGreetings  
    = Observable.create(new AlienGreetingOnSubscribe());
```

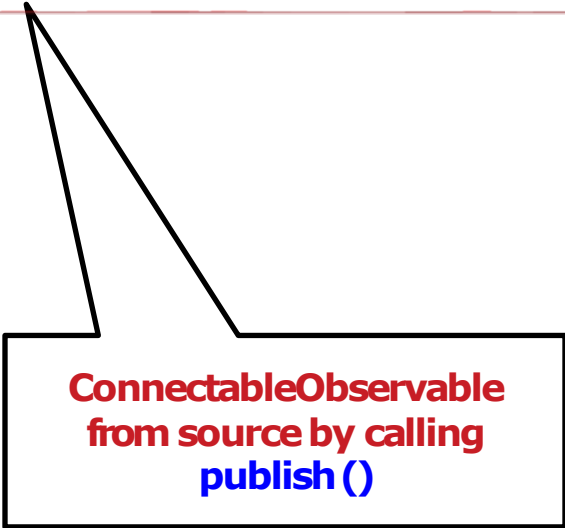


An Observable  
that emits  
Greetings from  
Earth

# Connectable Observable

```
Observable<String> earthGreetings  
    = Observable.create(new AlienGreetingOnSubscribe());
```

```
ConnectableObservable<String> hotGreetingObservable  
    = earthGreetings.publish();
```



**ConnectableObservable**  
from source by calling  
**publish()**

# Connectable Observable

```
Observable<String> earthGreetings  
    = Observable.create(new AlienGreetingOnSubscribe());  
  
ConnectableObservable<String> hotGreetingObservable  
    = earthGreetings.publish();  
  
hotGreetingObservable.connect();
```

**connect()** makes the Observable  
**HOT!** Starts emitting greetings.

# Connectable Observable

```
Observable<String> earthGreetings
    = Observable.create(new AlienGreetingOnSubscribe());

ConnectableObservable<String> hotGreetingObservable
    = earthGreetings.publish();

hotGreetingObservable.connect();

Subscription firstAlienShip = hotGreetingObservable
    .subscribe(t -> LOG.debug("First Alien Ship Received:" + t));
```



FirstAlien Ship  
subscribes for  
messages.

# Connectable Observable

```
Observable<String> earthGreeting
    = Observable.create(new AlienGreetingOnSubscribe());

ConnectableObservable<String> hotGreetingObservable
    = earthGreeting.publish();

hotGreetingObservable.connect();

Subscription firstAlienShip = hotGreetingObservable
    .subscribe(t -> LOG.debug("First Alien Ship Received:" + t));

Thread.sleep(200);
```

**Simulate time  
passing.**

# Connectable Observable

```
Observable<String> earthGreetings
    = Observable.create(new AlienGreetingOnSubscribe());


ConnectableObservable<String> hotGreetingObservable
    = earthGreetings.publish();

hotGreetingObservable.connect();

Subscription firstAlienShip = hotGreetingObservable
    .subscribe(t -> LOG.debug("First Alien Ship Received:" + t));

Thread.sleep(200);

Subscription secondAlienShip = hotGreetingObservable
    .subscribe(t -> LOG.debug("Second Alien Ship Received:" + t));
```



Second Alien Ship  
subscribes and  
receives  
messages.

# Console

```
Earth emitting:Hola 1439839860892
Earth emitting:Bonjorno 1439839861093
Earth emitting:Hello 1439839861292
First Alien Ship subscribing..
Earth emitting:Bonjorno 1439839861493
First Alien Ship Received :Bonjorno 1439839861493
Earth emitting:Hola 1439839861693
First Alien Ship Received :Hola 1439839861693
Earth emitting:Bonjorno 1439839861894
First Alien Ship Received :Bonjorno 1439839861894
Second Alient Ship subscribing...
Earth emitting:Bounjour 1439839862093
First Alien Ship Received :Bounjour 1439839862093
Second Alien Ship Received:Bounjour 1439839862093
Earth emitting:Hola 1439839862293
First Alien Ship Received :Hola 1439839862293
Second Alien Ship Received:Hola 1439839862293
Earth emitting:Hola 1439839862493
First Alien Ship Received :Hola 1439839862493
Second Alien Ship Received:Hola 1439839862493
Earth emitting:Namaste 1439839862694
First Alien Ship Received :Namaste 1439839862694
Second Alien Ship Received:Namaste 1439839862694
```

No  
Subscribers

First  
Observer

Second  
Observer

## ConnectableObservable – replay()


```
Observable<String> earthGreetings
    = Observable.create(new AlienGreetingOnSubscribe());

ConnectableObservable<String> hotGreetingObservable
    = earthGreetings.replay();

hotGreetingObservable.connect();

Thread.sleep(200); // Simulate a Delay

// Alien ship receives ALL greetings emitted
Subscription firstAlienShip = hotGreetingObservable
    .subscribe(t -> LOG.debug("First Alien Ship Received:" + t));
```



HOT!!! Starts emitting greetings.



# Console – replay()

```
Earth emitting:Namaste 1439840404012
Earth emitting:Bounjour 1439840404211
Earth emitting:Namaste 1439840404410
Earth emitting:Hello 1439840404611
Earth emitting:Bonjorno 1439840404810
Earth emitting:Namaste 1439840405010
First Alien Ship subscribing..
First Alien Ship Received:Namaste 1439840404012
First Alien Ship Received:Bounjour 1439840404211
First Alien Ship Received:Namaste 1439840404410
First Alien Ship Received:Hello 1439840404611
First Alien Ship Received:Bonjorno 1439840404810
First Alien Ship Received:Namaste 1439840405010
Earth emitting:Hola 1439840405210
First Alien Ship Received:Hola 1439840405210
```

No  
Subscribers

Observer gets it  
all

PublishSubject to create a Hot Observable

 **Another way to create a Hot Observable**

# Lab : Hot Observables