

Assign record values dynamically in Apex

Each standard and custom object inherits methods from the **SObject** class.

There are two methods worth considering in the context of our tip:

```
put(String fieldName, Object value)  
  
put(Schema.SObjectField field, Object value)
```

It can be very useful in Test Data Factory frameworks where values need to be overridden.



Dynamic assignment works perfectly with Maps, where the Field is a key. As mentioned on the previous slide, TDF can be a perfect use case for **put**.

```
public void updateFields(SObject record, Map<SObjectField, Object> fieldToValue) {  
    for (SObjectField field : fieldToValue.keySet()) {  
        record.put(field, fieldToValue.get(field));  
    }  
}
```

```
TDF.create(Account.SObjectType, new Map<SObjectField, Object>{  
    Account.Name => 'My Account',  
    Account.Industry => 'IT'  
});
```

```
TDF.create(Account.SObjectType, new Map<SObjectField, Object>{  
    'Name' => 'My Account',  
    'Industry' => 'IT'  
});
```