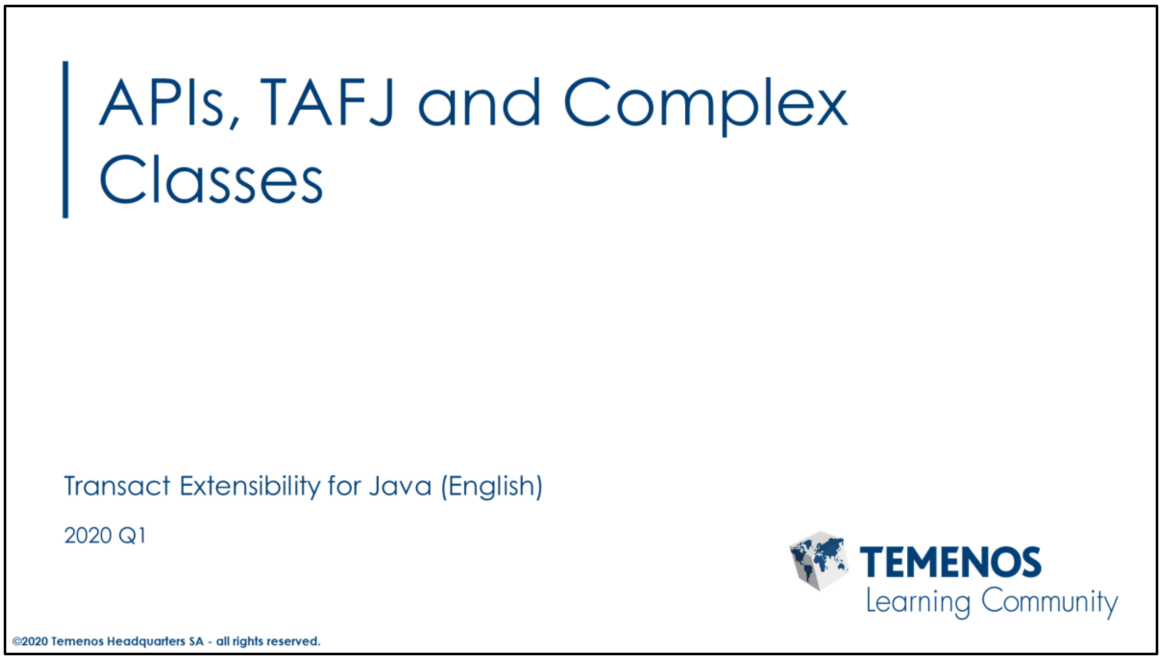


APIs, TAFJ and Complex Classes

Transact Extensibility for Java (English)

2020 Q1

TEMENOS
Learning Community

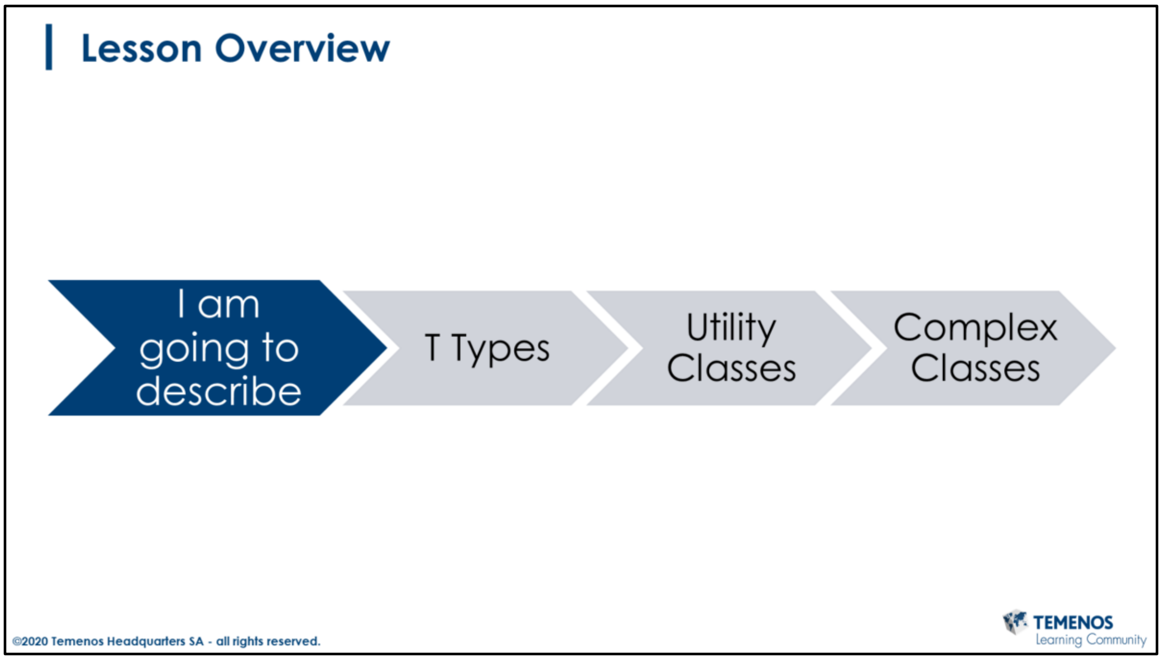
©2020 Temenos Headquarters SA - all rights reserved.

Copyright Notice

©2020 Temenos Headquarters SA - all rights reserved.

Warning: This document is protected by copyright law and international treaties. Unauthorised reproduction of this document, or any portion of it, may result in severe and criminal penalties, and will be prosecuted to the maximum extent possible under law.

TEMENOS
Learning Community



Lesson Overview

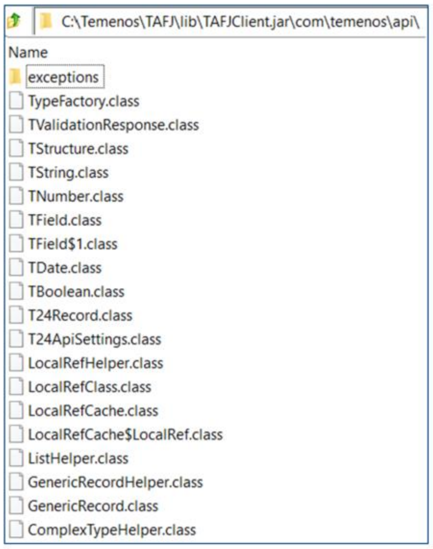
I am going to describe → T Types → Utility Classes → Complex Classes

©2020 Temenos Headquarters SA - all rights reserved.

TEMENOS Learning Community

'T' Types Classes

- Classes introduced by Temenos for Java Developers in TAFJClient.jar
- The reason for the T Types is to make them mutable. i.e. String is immutable



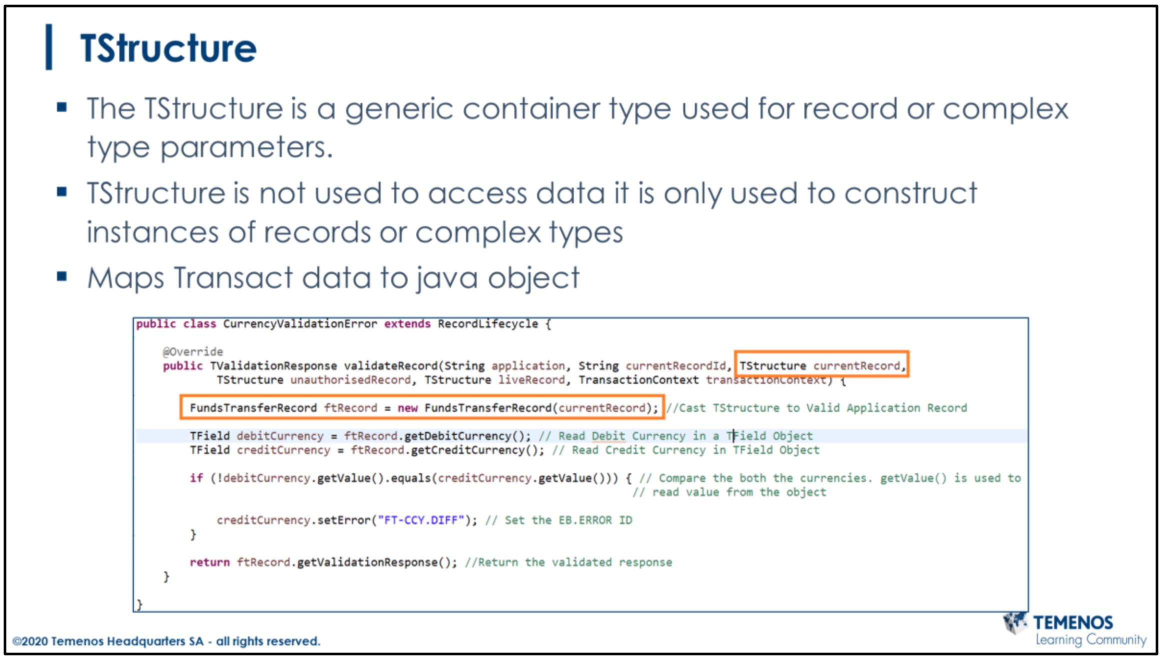
C:\Temenos\TAFJ\lib\TAFJClient.jar\com\temenos\api\

```
Name
exceptions
TypeFactory.class
TValidationResponse.class
TStructure.class
TString.class
TNumber.class
TField.class
TField$1.class
TDate.class
TBoolean.class
T24Record.class
T24ApiSettings.class
LocalRefHelper.class
LocalRefClass.class
LocalRefCache.class
LocalRefCache$LocalRef.class
ListHelper.class
GenericRecordHelper.class
GenericRecord.class
ComplexTypeHelper.class
```

©2020 Temenos Headquarters SA - all rights reserved.

TEMENOS Learning Community

'T' types are new classes introduced by Temenos for Java developers. These are available under TAFJClient.jar
This JAR can be found under TAFJ/Lib folder
The classes in the jar resides in TAFJ Framework Layer.
The reason for the T Types is to make them mutable. i.e. String is immutable



TStructure

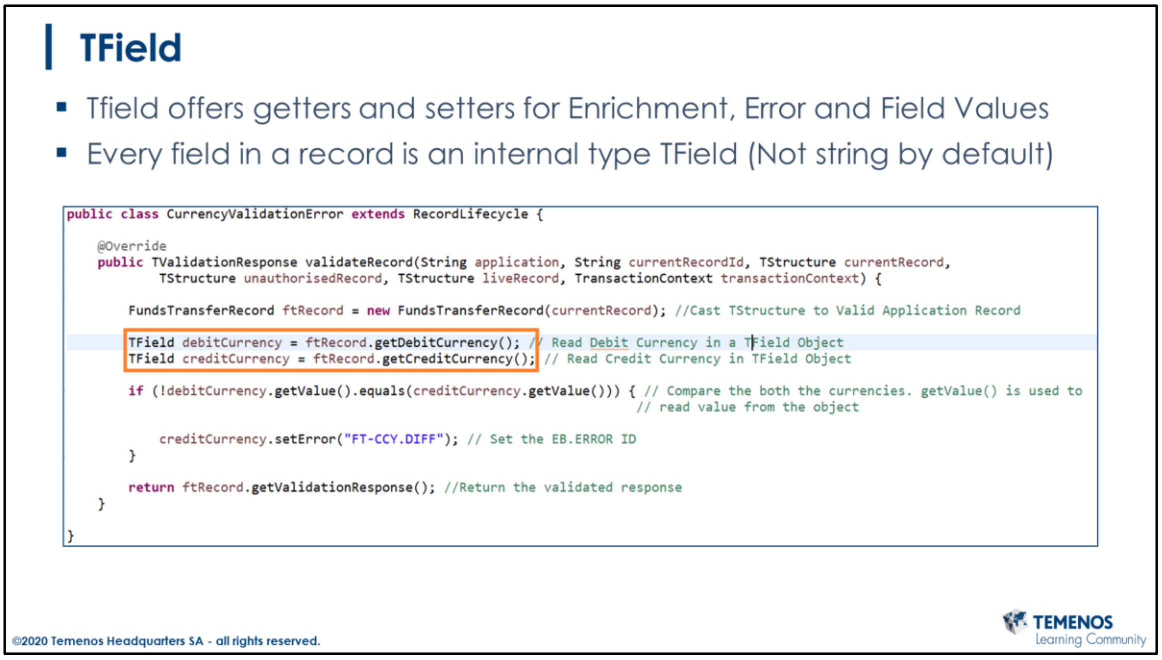
- The TStructure is a generic container type used for record or complex type parameters.
- TStructure is not used to access data it is only used to construct instances of records or complex types
- Maps Transact data to java object

```java
public class CurrencyValidationError extends RecordLifecycle {

    @Override
    public TValidationResponse validateRecord(String application, String currentRecordId, TStructure currentRecord,
            TStructure unauthorisedRecord, TStructure liveRecord, TransactionContext transactionContext) {

        FundsTransferRecord ftRecord = new FundsTransferRecord(currentRecord); //Cast TStructure to Valid Application Record

        TField debitCurrency = ftRecord.getDebitCurrency(); // Read Debit Currency in a TField Object
        TField creditCurrency = ftRecord.getCreditCurrency(); // Read Credit Currency in TField Object

        if (!debitCurrency.getValue().equals(creditCurrency.getValue())) { // Compare the both the currencies. getValue() is used to
                                                                          // read value from the object

            creditCurrency.setError("FT-CCY.DIFF"); // Set the EB.ERROR ID
        }

        return ftRecord.getValidationResponse(); //Return the validated response
    }

}
```

©2020 Temenos Headquarters SA - all rights reserved.

TEMENOS Learning Community

The TStructure is a generic container type used for record or complex type parameters.
TStructure is not used to access data it is only used to construct instances of records or complex types
Maps Transact data to java object



TField

- Tfield offers getters and setters for Enrichment, Error and Field Values
- Every field in a record is an internal type TField (Not string by default)

```
public class CurrencyValidationError extends RecordLifecycle {

    @Override
    public TValidationResponse validateRecord(String application, String currentRecordId, TStructure currentRecord,
            TStructure unauthorisedRecord, TStructure liveRecord, TransactionContext transactionContext) {

        FundsTransferRecord ftRecord = new FundsTransferRecord(currentRecord); //Cast TStructure to Valid Application Record

        TField debitCurrency = ftRecord.getDebitCurrency(); // Read Debit Currency in a TField Object
        TField creditCurrency = ftRecord.getCreditCurrency(); // Read Credit Currency in TField Object

        if (!debitCurrency.getValue().equals(creditCurrency.getValue())) { // Compare the both the currencies. getValue() is used to
                                                                          // read value from the object

            creditCurrency.setError("FT-CCY.DIFF"); // Set the EB.ERROR ID
        }

        return ftRecord.getValidationResponse(); //Return the validated response
    }
}
```

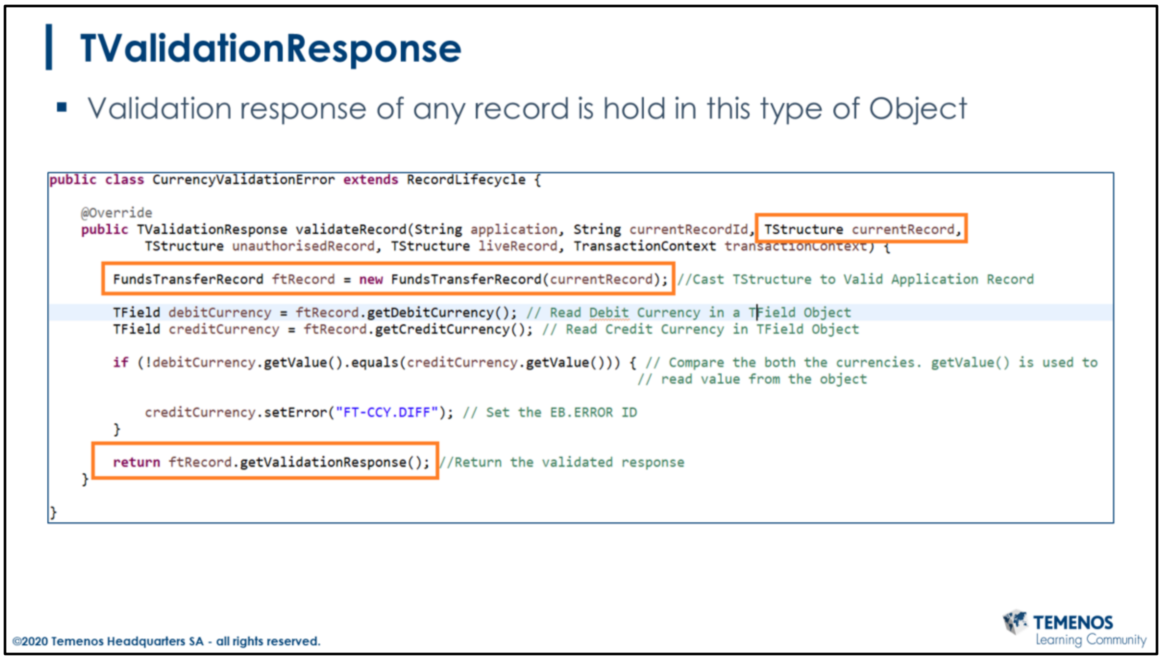
©2020 Temenos Headquarters SA - all rights reserved.

TEMENOS Learning Community

TField has getter and setter methods for field values, error as well as enrichments.

TField f1 = fr.getDebitCurrency(); → get the value from field DEBIT.CURRENCY of FUNDS.TRANSFER.
Setter can be used to set any value.

From the above e.g. f2.setError();→ Setting the Error message if DEBIT.CURRENCY and CREDIT.CURRENCY are not equal.



TValidationResponse

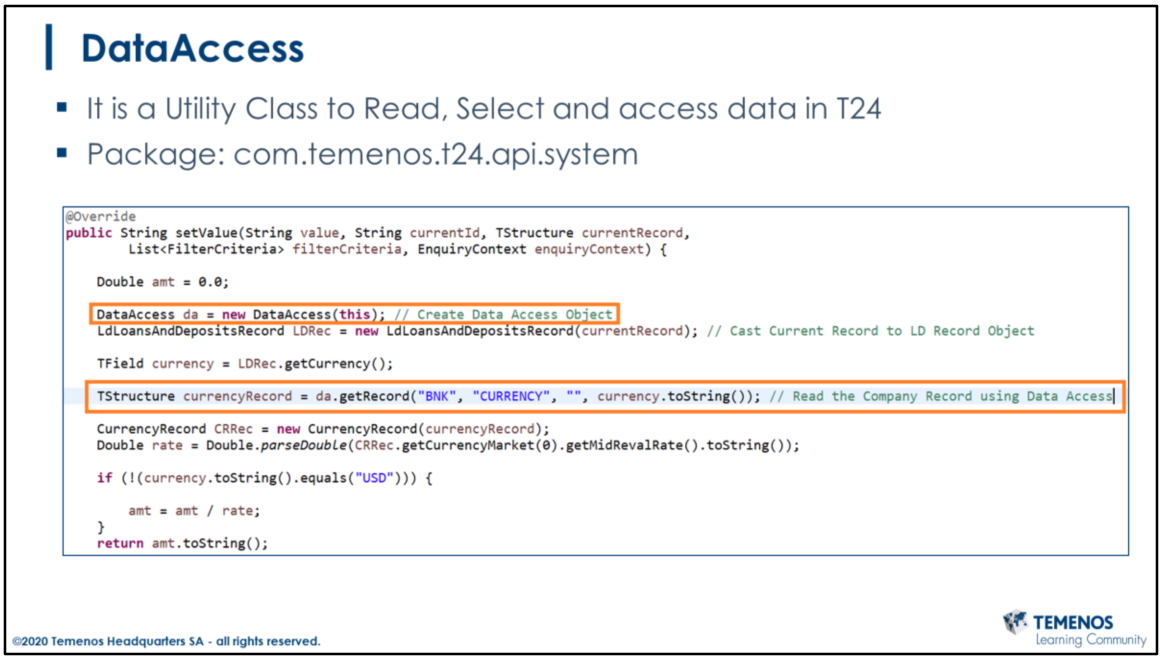
- Validation response of any record is hold in this type of Object

```
public class CurrencyValidationError extends RecordLifecycle {

    @Override
    public TValidationResponse validateRecord(String application, String currentRecordId, TStructure currentRecord,
            TStructure unauthorisedRecord, TStructure liveRecord, TransactionContext transactionContext) {

        FundsTransferRecord ftRecord = new FundsTransferRecord(currentRecord); //Cast TStructure to Valid Application Record

        TField debitCurrency = ftRecord.getDebitCurrency(); // Read Debit Currency in a TField Object
        TField creditCurrency = ftRecord.getCreditCurrency(); // Read Credit Currency in TField Object

        if (!debitCurrency.getValue().equals(creditCurrency.getValue())) { // Compare the both the currencies. getValue() is used to
                                                                          // read value from the object

            creditCurrency.setError("FT-CCY.DIFF"); // Set the EB.ERROR ID
        }

        return ftRecord.getValidationResponse(); //Return the validated response
    }

}
```

©2020 Temenos Headquarters SA - all rights reserved.

TEMENOS Learning Community

In above e.g. we are using TField to get the values of the fields from FUNDS.TRANSFER record . If debit and credit currency are not the same, error is set and response is returned using getValidationResponse();



DataAccess

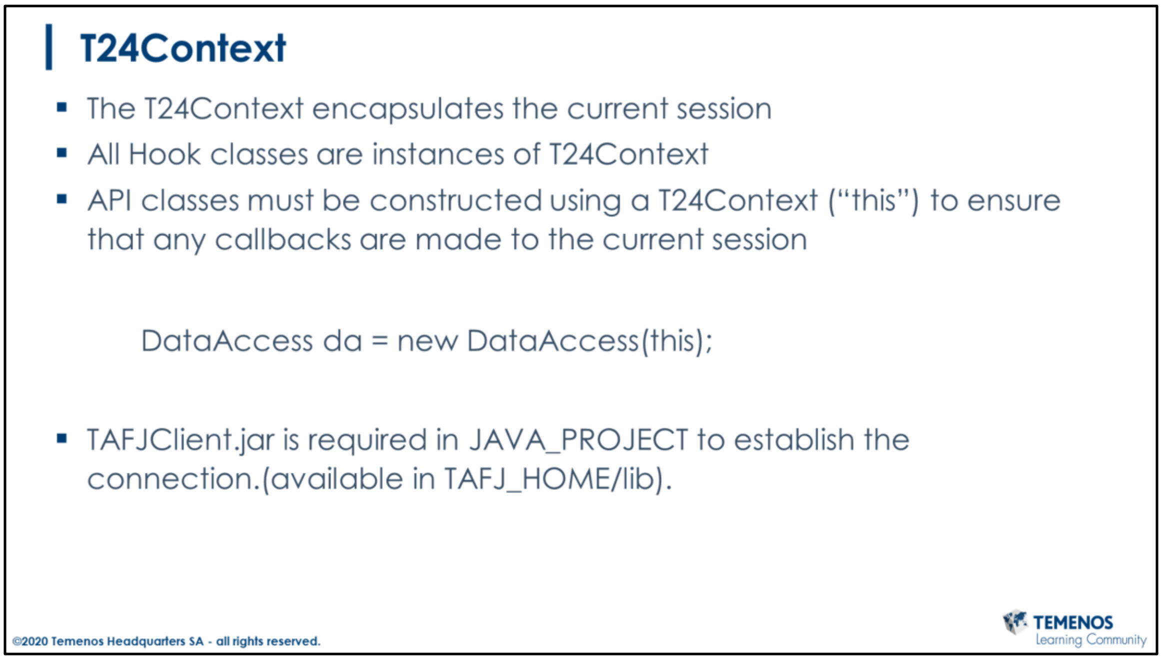
- It is a Utility Class to Read, Select and access data in T24
- Package: com.temenos.t24.api.system

```
@Override
public String setValue(String value, String currentId, TStructure currentRecord,
        List<FilterCriteria> filterCriteria, EnquiryContext enquiryContext) {

    Double amt = 0.0;

    DataAccess da = new DataAccess(this); // Create Data Access Object
    LdLoansAndDepositsRecord LDRec = new LdLoansAndDepositsRecord(currentRecord); // Cast Current Record to LD Record Object

    TField currency = LDRec.getCurrency();

    TStructure currencyRecord = da.getRecord("BNK", "CURRENCY", "", currency.toString()); // Read the Company Record using Data Access

    CurrencyRecord CRRec = new CurrencyRecord(currencyRecord);
    Double rate = Double.parseDouble(CRRec.getCurrencyMarket(0).getMidRevalRate().toString());

    if (!(currency.toString().equals("USD"))) {

        amt = amt / rate;
    }
    return amt.toString();
```

©2020 Temenos Headquarters SA - all rights reserved.

TEMENOS Learning Community

The system.DataAccess class is a utility class to read, select and access data in T24.
The api.records package is used to hold records from T24 applications


T24Context

- The T24Context encapsulates the current session
- All Hook classes are instances of T24Context
- API classes must be constructed using a T24Context (“this”) to ensure that any callbacks are made to the current session

DataAccess da = new DataAccess(this);

- TAFJClient.jar is required in JAVA_PROJECT to establish the connection.(available in TAFJ_HOME/lib).

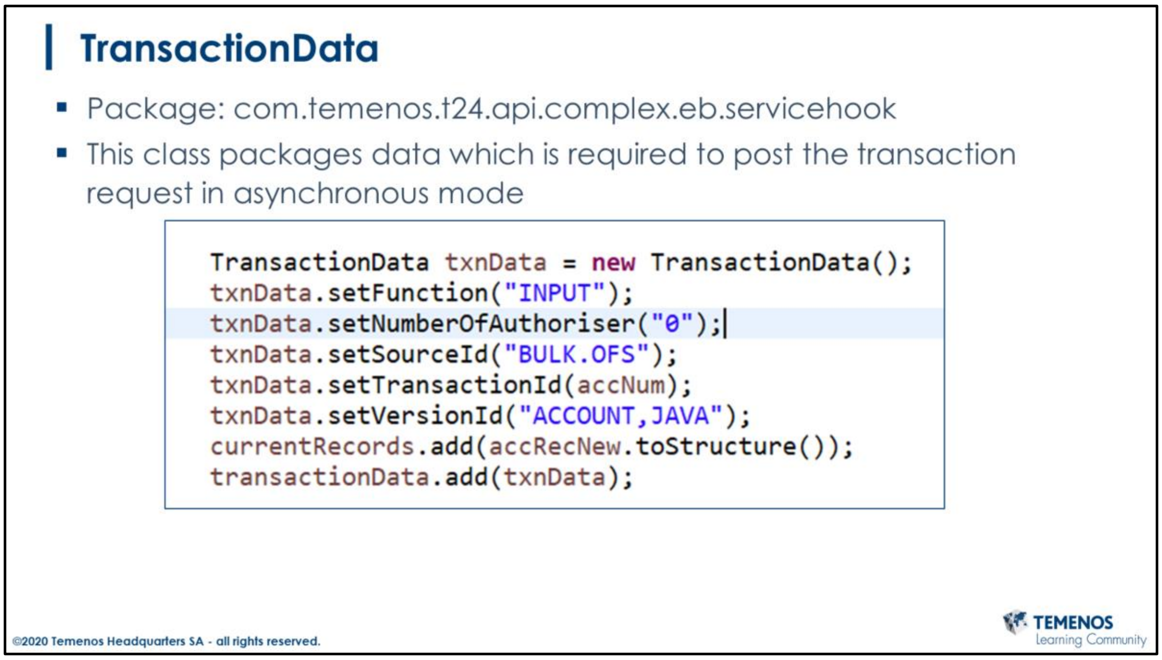
©2020 Temenos Headquarters SA - all rights reserved.

TEMENOS Learning Community

The T24Context encapsulates the current session
All Hook classes are instances of T24Context
API classes must be constructed using a T24Context (“this”) to ensure that any callbacks are made to the current session

DataAccess da = new DataAccess(this);

TAFJClient.jar is required in JAVA_PROJECT to establish the connection.(available in TAFJ_HOME/lib).



TransactionData

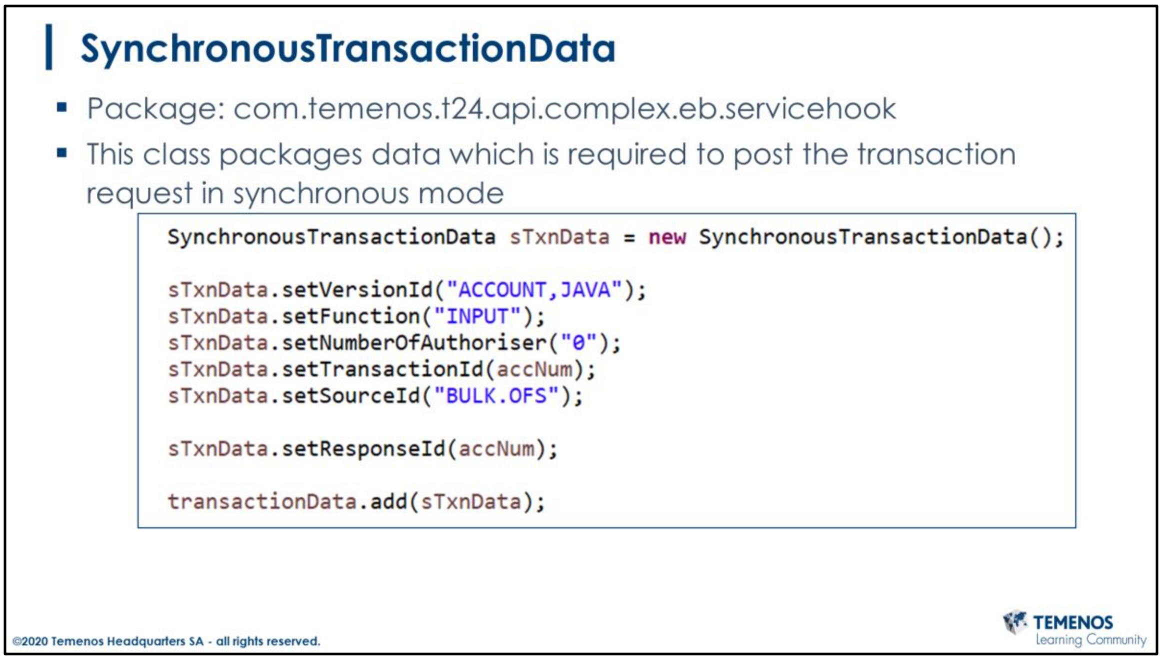
- Package: com.temenos.t24.api.complex.eb.servicehook
- This class packages data which is required to post the transaction request in asynchronous mode

```
TransactionData txnData = new TransactionData();
txnData.setFunction("INPUT");
txnData.setNumberOfAuthoriser("0");
txnData.setSourceId("BULK.OFS");
txnData.setTransactionId(accNum);
txnData.setVersionId("ACCOUNT,JAVA");
currentRecords.add(accRecNew.toStructure());
transactionData.add(txnData);
```

©2020 Temenos Headquarters SA - all rights reserved.

TEMENOS Learning Community

This class packages data which is required to post the transaction request in asynchronous mode



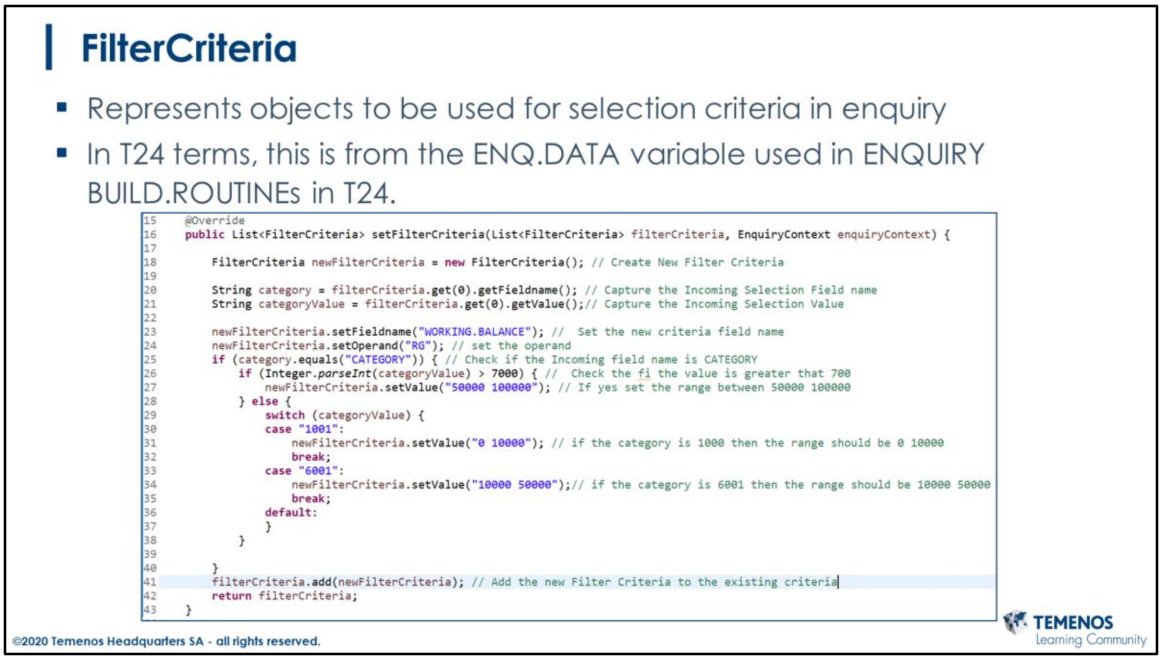
SynchronousTransactionData

- Package: com.temenos.t24.api.complex.eb.servicehook
- This class packages data which is required to post the transaction request in synchronous mode

```
SynchronousTransactionData sTxnData = new SynchronousTransactionData();

sTxnData.setVersionId("ACCOUNT,JAVA");
sTxnData.setFunction("INPUT");
sTxnData.setNumberOfAuthoriser("0");
sTxnData.setTransactionId(accNum);
sTxnData.setSourceId("BULK.OFS");

sTxnData.setResponseId(accNum);

transactionData.add(sTxnData);
```

©2020 Temenos Headquarters SA - all rights reserved.

This class packages data which is required to post the transaction request in asynchronous mode



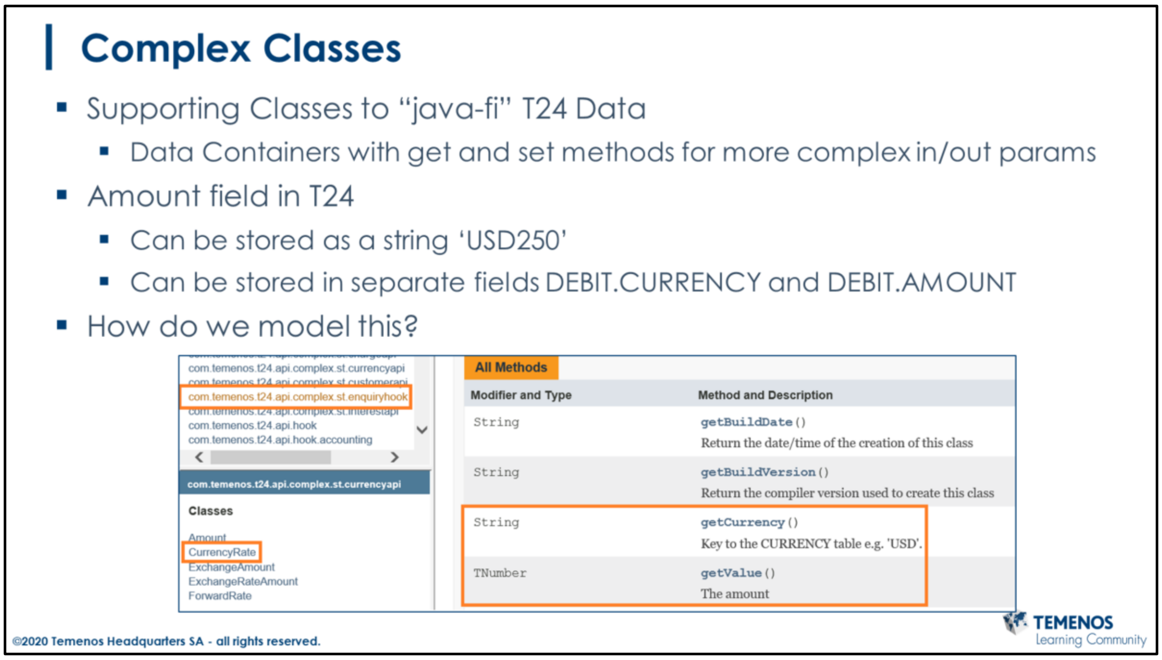
FilterCriteria

- Represents objects to be used for selection criteria in enquiry
- In T24 terms, this is from the ENQ.DATA variable used in ENQUIRY BUILD.ROUTINEs in T24.

```java
@Override
public List<FilterCriteria> setFilterCriteria(List<FilterCriteria> filterCriteria, EnquiryContext enquiryContext) {

    FilterCriteria newFilterCriteria = new FilterCriteria(); // Create New Filter Criteria

    String category = filterCriteria.get(0).getFieldname(); // Capture the Incoming Selection Field name
    String categoryValue = filterCriteria.get(0).getValue();// Capture the Incoming Selection Value

    newFilterCriteria.setFieldname("WORKING.BALANCE"); //  Set the new criteria field name
    newFilterCriteria.setOperand("RG"); // set the operand
    if (category.equals("CATEGORY")) { // Check if the Incoming field name is CATEGORY
        if (Integer.parseInt(categoryValue) > 7000) { //  Check the fi the value is greater that 700
            newFilterCriteria.setValue("50000 100000"); // If yes set the range between 50000 100000
        } else {
            switch (categoryValue) {
            case "1001":
                newFilterCriteria.setValue("0 10000"); // if the category is 1000 then the range should be 0 10000
                break;
            case "6001":
                newFilterCriteria.setValue("10000 50000");// if the category is 6001 then the range should be 10000 50000
                break;
            default:
            }
        }

    }
    filterCriteria.add(newFilterCriteria); // Add the new Filter Criteria to the existing criteria
    return filterCriteria;
}
```

©2020 Temenos Headquarters SA - all rights reserved.

Represents objects to be used for selection criteria in enquiry
In T24 terms, this is from the ENQ.DATA variable used in ENQUIRY BUILD.ROUTINEs in T24.



Complex Classes

- Supporting Classes to "java-fi" T24 Data
  - Data Containers with get and set methods for more complex in/out params
- Amount field in T24
  - Can be stored as a string 'USD250'
  - Can be stored in separate fields DEBIT.CURRENCY and DEBIT.AMOUNT
- How do we model this?

com.temenos.t24.api.complex.st.currencyapi
com.temenos.t24.api.complex.st.customerapi
com.temenos.t24.api.complex.st.enquiryhook
com.temenos.t24.api.complex.st.interestapi
com.temenos.t24.api.hook
com.temenos.t24.api.hook.accounting

com.temenos.t24.api.complex.st.currencyapi

Classes

Amount
CurrencyRate
ExchangeAmount
ExchangeRateAmount
ForwardRate

All Methods

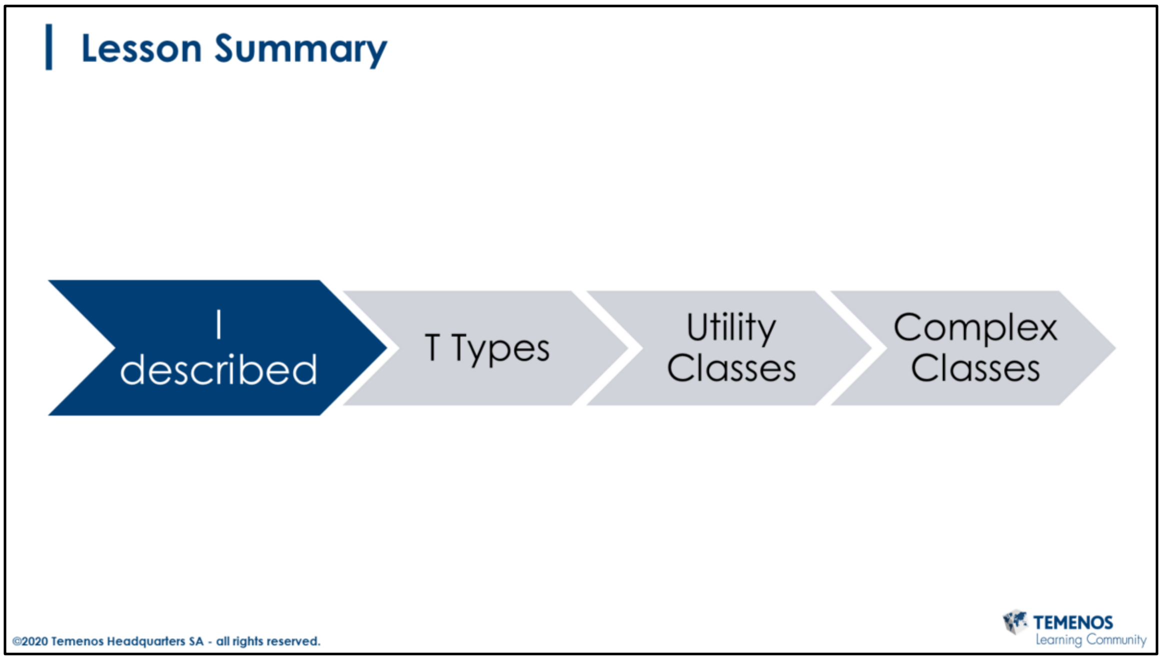
Modifier and Type	Method and Description
`String`	`getBuildDate()` Return the date/time of the creation of this class
`String`	`getBuildVersion()` Return the compiler version used to create this class
`String`	`getCurrency()` Key to the CURRENCY table e.g. 'USD'.
`TNumber`	`getValue()` The amount

©2020 Temenos Headquarters SA - all rights reserved.

TEMENOS Learning Community

A Complex class is a convenient way of **grouping** several pieces of related information together. For e.g amount is represented as USD250 – consists of a Currency as well as a value.

For e.g. the values(field name, operand, value) in a dynamic selection box of an ENQUIRY is a set of related data (but not a record) and a complex class is used to hold the values. **com.temenos.t24.api.complex.eb.enquiryhook.FilterCriteria**



Lesson Summary

I described
T Types
Utility Classes
Complex Classes

©2020 Temenos Headquarters SA - all rights reserved.

TEMENOS Learning Community

TEMENOS
Learning Community

thank.you

tlc.temenos.com