coreservlets.com – Hadoop Course

HBase Java Advanced API

In this exercise, you will have a chance to develop code based on HBase Java API. Your code will retrieve sets of rows utilizing Scan API.

Approx. Time: 45 minutes

Provided

You will find *Exercise_Advanced_Book* HBase table with the following data:

Row ID	info:title	into:description	author:first	author:last
1	Faster than the speed love	Long book about love.	Brian	Dog
2	Long day	Story about Monday.	Emily	Blue
3	Flying Car	Novel about airplanes.	Phil	High

Perform

Write Java code that will exercise Htable's API. Utilize Scan API against *Exercise_Advanced_Book* table to perform the following tasks:

1. Display all the records to the screen for the Book table (hint: Scan through the records)
2. Display title and description for the first 2 records (hint: Scan through the records)
3. Display cells which contain "*about*" word using Filters.
4. Retrieve row ids only using Filters.

Expected Output

Your output should be similar to something like this:

```
------------------------------
1: All results are:
Result with rowId [1], title=Faster than the speed love, description=Long book
about love., first name=Brian, last name=Dog
Result with rowId [2], title=Long day, description=Story about Monday., first
name=Emily, last name=Blue
Result with rowId [3], title=Flying Car, description=Novel about airplanes., first
name=Phil, last name=High
------------------------------
2: First 2 results
Result with rowId [1], title=Faster than the speed love, description=Long book
about love.
Result with rowId [2], title=Long day, description=Story about Monday.
------------------------------
3: Results with filter
Result with rowId [1], description=Long book about love.
```

```
Result with rowId [2], description=Story about Monday.

Result with rowId [3], description=Novel about airplanes.

------------------------------

4: Only Row Ids

1

2

3

------------------------------
```

Hints/Suggestions

1. Re-use Configuration and HTable instances. Why is that a good idea?
2. Write all the tasks in the same Java file.
3. Create a method to print a single Result instance.
4. Use *static* construct for all the schema names such as table name, families and even columns. Create these as byte arrays to avoid unnecessary conversions.

Solution

1. The code can be found in the Solutions project:

   `hbase.javaAPIAdvanced.JavaAPIAdvancedSolution.java`

2. To execute the solution

   ```
   $ yarn jar $PLAY_AREA/Solutions.jar hbase.javaAPIAdvanced.JavaAPIAdvancedSolution
   ```