To accomplish the tasks you've outlined using HBase, follow these steps:

### 1. Create HBase Table

First, create a table named `employees` with two column families: `personal` for personal details and `professional` for professional details.

```shell

create 'employees', 'personal', 'professional'

```

### 2. Describe the Table

After inserting all rows of data into it, describe the table to see its structure and contents.

```shell

```

### 3. Update Salary

To update the salary of an employee with `empid` 3 from 55000 to 65000, use the `put` command.

```shell

```

Then, describe the table again to show the updates.

```shell

describe 'employees'

```

### 4. Retrieve Employees Details

To retrieve employees whose salary is greater than or equals to 70000, you can use a scan with a filter.

```shell

scan 'employees', {FILTER => "SingleColumnValueFilter('professional', 'salary', =, 'binary:70000')"}

```

### 5. Read Personal Data of David

To read the personal data of an employee named David, use the `get` command.

```shell

get 'employees', 'David', {COLUMN => 'personal:name'}

```

### 6. Describe Employee Details

To describe employee details whose designation is a data analyst, you would typically need to scan the table and filter by the designation. However, HBase does not support direct queries like SQL databases. You would need to scan the table and filter the results in your application.

### 7. Count Rows and Columns

HBase does not provide a direct command to count the number of rows and columns. You would need to scan the table and count the rows and columns in your application.

### 8. Delete Age Column

To delete the age column from personal data, you would use the `deleteall` command. However, this command deletes all versions of the specified column.

```shell

```

hbase:002:0> describe 'employees'

Table employees is ENABLED

employees, {TABLE\_ATTRIBUTES => {METADATA => {'hbase.store.file-tracker.impl' => 'DEFAULT'}}}

COLUMN FAMILIES DESCRIPTION

{NAME => 'personal', INDEX\_BLOCK\_ENCODING => 'NONE', VERSIONS => '1', KEEP\_DELETED\_CELLS => 'FALSE', DATA\_BLOCK\_ENCODING => 'NONE', TTL => 'FOREVER', MIN\_VERSIONS => '0', REPLICATION\_SCOPE => '0', BLOOMF

ILTER => 'ROW', IN\_MEMORY => 'false', COMPRESSION => 'NONE', BLOCKCACHE => 'true', BLOCKSIZE => '65536 B (64KB)'}

{NAME => 'professional', INDEX\_BLOCK\_ENCODING => 'NONE', VERSIONS => '1', KEEP\_DELETED\_CELLS => 'FALSE', DATA\_BLOCK\_ENCODING => 'NONE', TTL => 'FOREVER', MIN\_VERSIONS => '0', REPLICATION\_SCOPE => '0', BL

OOMFILTER => 'ROW', IN\_MEMORY => 'false', COMPRESSION => 'NONE', BLOCKCACHE => 'true', BLOCKSIZE => '65536 B (64KB)'}