

[illegible]

Task 2

Team T-18: Popular Names

Task Zero

Reporting 1

Reporting 2

Reporting 3

Application1

Application 2

Application 3

Generate a brief report in response to the queries on the popularity of a particular name over a given period.

Name

David

Gender:

☒ Male

☐ Female

Period (1880 - 2019):

1941

to

1945

REPORT

Console

In the year 1945, the number of birth with name David is 37093, which represents 2.76 percent of total Male births in 1945.
The year when the name David was most popular is 1945.
In that year, the number of births is 37093, which represents 2.76 percent of the total M birth in 1945.

Report

Year	Rank	Count	Percentage	
1941	7	30551	2.4879859862973874	
1942	6	35892	2.5994116343442157	
1943	6	37237	2.609742046443695	
1944	6	37374	2.7429351475720245	
1945	6	37093	2.7565936638401913	

Task 3

Team T-18: Popular Names

Task ZeroReporting 1Reporting 2Reporting 3Application 1Application 2Application 3

Generate a brief report in response to the queries on identifying the names that have maintained a high level of popularity within Top N over a given period.

Top N (N>=1): Gender: ☒ Male ☐ Female

Period (1880 - 2019): to

REPORT

Console

Over the period 1941 to 1945, the 9 names below maintained a high level of popularity within Top 10.
James
Robert
John
William
Richard
Charles
David
Thomas
Ronald

Report

Name	Lowest Rank [in year]	Highest Rank [in year]	Trend	
James	1 [1941]	1 [1941]	FLAT	
Robert	2 [1941]	2 [1941]	FLAT	
John	3 [1941]	3 [1941]	FLAT	
William	4 [1941]	4 [1941]	FLAT	
Richard	5 [1941]	5 [1941]	FLAT	
Charles	7 [1942]	6 [1941]	DOWN	
David	7 [1941]	6 [1942]	UP	
Thomas	8 [1941]	8 [1941]	FLAT	
Ronald	10 [1944]	9 [1941]	DOWN	

Task 4

Team T-18: Popular Names

Task ZeroReporting 1Reporting 2Reporting 3Application 1Application 2Application 3

Recommend a baby's name given information of the parents.
All years should be from 1880 to 2019.

Dad's name: Dad's year of birth:

Mom's name: Mom's year of birth:

☒ With vintage year

☐ Without vintage year

Get Recommendation

Console

Boy name: Joshua
Girl name: Amanda

Task 5

Team T-18: Popular Names

Task Zero

Reporting 1

Reporting 2

Reporting 3

Application1

Application 2

Application 3

What is your name?

David

Which year were you born?

2000

What is your gender?

☒ Male

☐ Female

Which gender of your mate do you prefer?

☐ Male

☒ Female

Do you prefer younger or older mate?

☒ Younger

☐ Older

Get Recommendation

Console

Recommended mate name: Grace

Task 6

Team T-18: Popular Names

Task Zero

Reporting 1

Reporting 2

Reporting 3

Application1

Application 2

Application 3

Predicting scores for compatible pairs (0%: Not Compatible; 100%: Perfect Match).
Both your birth year your preferred mate's birth year should be from 1880 to 2019.

My name:

David

My gender:

☒ Male

☐ Female

My year of birth (1880 - 2019):

2000

My Mate's name:

John

My Mate's gender:

☒ Male

☐ Female

My Preferred Mate Age:

☒ Younger

☐ Older

Get Score

Console

The compatible score for David and John is 87.500000 %.

2. Report on the unit testing for the implemented tasks (100% pass)

[In **test** file on Repo]

Test Summary

28	0	0	34.057s	100%
tests	failures	ignored	duration	successful

Packages

Classes

Package	Tests	Failures	Ignored	Duration	Success rate
comp3111.popnames	28	0	0	34.057s	100%

Generated by [Gradle 6.3](#) at Dec 4, 2020, 8:30:51 PM

3. Report on the coverage test (>65% branch coverage)

[In **jacocoHTML** file on Repo]

PopNames3111 Sessions

PopNames3111

Element	Missed Instructions	Cov.	Missed Branches	Cov.	Missed	Cxty	Missed	Lines	Missed	Methods	Missed	Classes
comp3111.popnames	<div><div></div></div>	79%	<div><div></div></div>	69%	57	127	96	434	10	42	1	6
Total	470 of 2,277	79%	52 of 170	69%	57	127	96	434	10	42	1	6

Created with JaCoCo 0.8.5.201910111838

4. Documentation on the implemented tasks using JavaDoc

[In **javadoc** file on Repo]

Part of the documentation is shown here.

PACKAGE	CLASS	TREE	DEPRECATED	INDEX	HELP
SEARCH: <input type="text"/>					
Package comp3111.popnames					
Class Summary					
Class	Description				
AnalyzeNames	This class provide methods to access the database and deal with the data, and serve as auxiliary fuctions for functions in the controller class.				
Controller	This is the controller class that deals with initial input.				
MyApplication	The entry point of the entire program.				
TableWrapper	This class aids with the display of the result.				

PACKAGE	CLASS	TREE	DEPRECATED	INDEX	HELP
---------	-------	------	------------	-------	------

The screenshots of two classes' documentation is shown here as an illustration.

AnalyzeNames Class

PACKAGE	CLASS	TREE	DEPRECATED	INDEX	HELP
SUMMARY: NESTED FIELD CONSTR METHOD DETAIL: FIELD CONSTR METHOD SEARCH: <input type="text"/>					
Class AnalyzeNames					
java.lang.Object comp3111.popnames.AnalyzeNames					
public class AnalyzeNames extends java.lang.Object					
This class provide methods to access the database and deal with the data, and serve as auxiliary fuctions for functions in the controller class.					
Field Summary					
Fields					
Modifier and Type	Field	Description			
static int	MAX_YEAR	Define the max supported year			
static int	MIN_YEAR	Define the min supported year			
Constructor Summary					
Constructors					
Constructor	Description				
AnalyzeNames()					

Method Summary		
All Methods	Static Methods	Concrete Methods
Modifier and Type		Description
static float	compatibleScore(java.lang.String iName, java.lang.String iGender, int iYOB, java.lang.String iNameMate, java.lang.String iGenderMate, int oYOB)	The function is for calculating score in Task 6
static org.apache.commons.csv.CSVParser	getFileParser(int year)	The function reads the data for a given year
static int[]	getHighAndLowRank(int yearFrom, int yearTo, java.lang.String name, java.lang.String gender)	The function is for get high And low rank during a period of time in Task 3
static java.util.Map.Entry<java.lang.String, java.lang.Integer>	getMaxNameCount(int yearFrom, int yearTo, java.lang.String gender)	The function is for get the compute the name with most top spots during a period of time in Task 1
static int	getMaxRank(int year, java.lang.String gender)	The function is for get the largest rank for a gender given a year
static int	getMaxRankInPeriod(int yearFrom, int yearTo, java.lang.String gender)	The function is for get the largest rank for a gender during a period of time
static int	getMostPopularYear(int yearFrom, int yearTo, java.lang.String name, java.lang.String gender)	The function is for get the most popular year during a period of time in Task 2
static java.lang.String	getName(int year, int rank, java.lang.String gender)	The function is for get the name of a given rank for a given gender in a given year
static int	getNameCount(int year, java.lang.String name, java.lang.String gender)	The function is for get the number of babies using a certain name in a certain year for Task 2
static double	getNamePercentage(int year, java.lang.String name, java.lang.String gender)	The function is to get the name percentage in a certain year in Task 2
static int	getRank(int year, java.lang.String name, java.lang.String gender)	The function is for get the name of a given rank for a given gender in a given year
static java.lang.String	getSummary(int year)	The function give a summary report for a given year
static javafx.util.Pair<java.lang.String, java.lang.String>	recommendBabyName(java.lang.String dadName, int dadYOB, java.lang.String momName, int momYOB, int vintageYear)	The function is for recommended BabyName in Task 4
static java.lang.String	recommendedMateName(java.lang.String yourName, int YOB, java.lang.String yourGender, java.lang.String mateGender, java.lang.String preference)	The function is for recommended MateName in Task 5
Methods inherited from class java.lang.Object		
clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait		

Field Details

MIN_YEAR
public static final int MIN_YEAR
Define the min supported year
See Also: Constant Field Values
MAX_YEAR
public static final int MAX_YEAR
Define the max supported year

Constructor Details

AnalyzeNames
public AnalyzeNames()

Method Details

getFileParser
public static org.apache.commons.csv.CSVParser getFileParser(int year)
The function reads the data for a given year
Parameters: year - The year of interest.
getSummary
public static java.lang.String getSummary(int year)
The function give a summary report for a given year
Parameters: year - The year of interest.

getMaxRank
<pre>public static int getMaxRank(int year, java.lang.String gender)</pre> <p>The function is for get the largest rank for a gender given a year</p> <p>Parameters: year - The year of interest. gender - The interested gender of the name.</p>
getMaxRankInPeriod
<pre>public static int getMaxRankInPeriod(int yearFrom, int yearTo, java.lang.String gender)</pre> <p>The function is for get the largest rank for a gender during a period of time</p> <p>Parameters: yearFrom - The starting year of the period. yearTo - The ending year of the period. gender - The interested gender of the name.</p>
getNameCount
<pre>public static int getNameCount(int year, java.lang.String name, java.lang.String gender)</pre> <p>The function is for get the number of babies using a certain name in a certain year for Task 2</p>

... (omitted several functions since it's only for illustration purpose)

recommendedMateName
<pre>public static java.lang.String recommendedMateName(java.lang.String yourName, int YOB, java.lang.String yourGender, java.lang.String mateGender, java.lang.String preference)</pre> <p>The function is for recommended MateName in Task 5</p> <p>Parameters: yourName - Name of the person seeking advices YOB - Year of Birth of the person seeking advices yourGender - Gender of the person seeking advices mateGender - Gender of the soulmate to be matched preference - Preference on having a younger or older soulmate</p>
compatibleScore
<pre>public static float compatibleScore(java.lang.String iName, java.lang.String iGender, int iYOB, java.lang.String iNameMate, java.lang.String iGenderMate, int oYOB)</pre> <p>The function is for calculating score in Task 6</p> <p>Parameters: iName - Name of the person seeking advices iGender - Gender of the person seeking advices iYOB - Year of Birth of the person seeking advices</p>

TableWrapper Class

Class TableWrapper

java.lang.Object
comp3111.popnames.TableWrapper

public class TableWrapper
extends java.lang.Object

This class aids with the display of the result.

Constructor Summary

Constructors

Constructor	Description
TableWrapper(java.util.List<java.lang.String> columnNames)	This is the constructor that initialize the columns of the table

Method Summary

All Methods

Instance Methods

Concrete Methods

Modifier and Type	Method	Description
void	addRow(java.util.List<java.lang.String> row)	This function helps to add a row to the table
javafx.scene.control.TableView<java.util.List<java.lang.String>>	getTable()	This function returns the table view

Methods inherited from class java.lang.Object

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

Constructor Details

TableWrapper

```
public TableWrapper(java.util.List<java.lang.String> columnNames)
```

This is the constructor that initialize the columns of the table

Parameters:

columnNames - The list of column names to be added to the table.

Method Details

addRow

```
public void addRow(java.util.List<java.lang.String> row)
```

This function helps to add a row to the table

Parameters:

row - The row to be added to the table

getTable

```
public javafx.scene.control.TableView<java.util.List<java.lang.String>> getTable()
```

This function returns the table view

5. Supplementary Notes

In task 6, we define the score as $1 - |(oRank - oRankMate)| / \max(oRank, oRankMate)$, which is in range (0,1].