Andrés Camilo Romero Ruiz – A00380637

Camilo Carmona Valencia – A00381090

Samuel Gutiérrez - A00381035

|  |  |
| --- | --- |
| Cliente | Desconocido |
| Usuario | Desconocido |
| Requerimientos funcionales | * R1. El programa debe permitirle al usuario ingresar comandos por consola que serán de alguno de los tres tipos: **AGREGAR DATOS**, **BÚSQUEDA Y FILTRADO**, **ORDENAMIENTO** y **ELIMINACIÓN** * R2. El programa debe permitirle al usuario agregar países que cuenten con los atributos id, name, population y countryCode. Además de ciudades con los atributos id, name, countryID y population bajo el formato INSERT INTO countries(id, name, population, countryCode) VALUES ('6ec3e8ec-3dd0-11ed-b878-0242ac120002', 'Colombia', 50.2, '+57') y INSERT INTO cities(id, name, countryID, population) VALUES ('e4aa04f6-3dd0-11ed-b878-0242ac120002', 'Cali', '6ec3e8ec-3dd0-11ed-b878-0242ac120002', 2.2) respectivamente   + R2.1 El programa debe arrojar una excepción si al agregar una ciudad el usuario ingresa un countryID no registrado, alguno de los datos o el comando está en el formato incorrecto * R3. El programa debe permitirle al usuario filtrar y buscar datos ya registrados haciendo uso de las palabras reservadas SELECT \* FROM cities/countries WHERE variable condicion donde WHERE es de carácter opcional * R4. El programa debe permitirle al usuario ordenar los datos ya registrados haciendo uso de las palabras reservadas SELECT \* FROM cities/countries WHERE variable condicion ORDER BY variable donde WHERE es de carácter opcional * R5. El programa debe permitirle al usuario borrar algún dato ya registrado haciendo uso de las palabras reservadas DELETE FROM cities/countries WHERE variable condicion * R6. El programa debe poder importar datos escritos en un archivo de formato Json cada que se inicie el programa * R7. El programa debe poder guardar los datos registrados en un archivo de formato Json cuando el usuario seleccione la opción salir, el programa guardará los datos y posterior a ello se cerrará * R8. El programa debe poder leer comandos escritos en un archivo de formato SQL y ejecutarlos, el archivo lo encontrará a través de un path dado por el usuario |
| Contexto del problema | Debemos desarrollar un programa en el cual se puedan almacenar los registros de ciudades y países ingresados por el usuario o importados a través de un archivo. Además hemos de permitir al usuario agregar, filtrar, ordenar y borrar registros siempre y cuando cumpla con la sintaxis de los comandos |

|  |  |  |  |
| --- | --- | --- | --- |
| Nombre o identificador | **R1: Lectura de Comandos** | | |
| Resumen | El sistema debe leer el comando ingresado por el usuario y validar que comience por INSERT INTO (agregar), SELECT \* FROM (filtrar u ordenar) o DELETE (borrar) | | |
| Entradas | **Nombre entrada** | **Tipo de dato** | **Condición de selección o repetición** |
| command | String | **Que el comando no comience con alguna de las palabras válidas** |
| Actividades generales necesarias para obtener los resultados | Que haya un comando escrito por consola | | |
| Resultado o Postcondición | Redirigir el comando a alguno de los verificadores de tipo para validar el resto de la sintaxis | | |
| Salidas | **Nombre salida** | **Tipo de dato** | **Condición de selección o repetición** |
| commandParts | String[] |  |
|  |  |  |

|  |  |  |  |
| --- | --- | --- | --- |
| Nombre o identificador | **R2: Validación de comando AGREGAR** | | |
| Resumen | El método debe asegurarse de que toda la sintaxis del comando de tipo agregar sea válida | | |
| Entradas | **Nombre entrada** | **Tipo de dato** | **Condición de selección o repetición** |
| commandParts | String[] |  |
| Actividades generales necesarias para obtener los resultados | Que el arreglo esté inicializado | | |
| Resultado o Postcondición | Redirigir las variables ingresadas a un verificador que se asegure de que para el caso de los países no haya id, name o countryCode repetidos, y para el caso de las ciudades que no haya un id repetido y que el countryID exista dentro de los registros | | |
| Salidas | **Nombre salida** | **Tipo de dato** | **Condición de selección o repetición** |
| values | String[] |  |
|  |  |  |

|  |  |  |  |
| --- | --- | --- | --- |
| Nombre o identificador | **R2.1: Validación de datos AGREGAR** | | |
| Resumen | El método debe asegurarse de que los datos con los que se va a crear la ciudad o país sean válidos | | |
| Entradas | **Nombre entrada** | **Tipo de dato** | **Condición de selección o repetición** |
| values | String[] |  |
| Actividades generales necesarias para obtener los resultados | Que el arreglo esté inicializado | | |
| Resultado o Postcondición | Crear el objeto de tipo país o ciudad si los valores son válidos o en caso contrario no añadir nada, en ambos se enviará un mensaje de confirmación o error respectivamente | | |
| Salidas | **Nombre salida** | **Tipo de dato** | **Condición de selección o repetición** |
| msg (mensaje de confirmación o error) | String |  |
|  |  |  |

|  |  |  |  |
| --- | --- | --- | --- |
| Nombre o identificador | **R3: Validación de comando BÚSQUEDA Y FILTRADO** | | |
| Resumen | El método debe asegurarse de que toda la sintaxis del comando de tipo búsqueda y filtrado sea válida | | |
| Entradas | **Nombre entrada** | **Tipo de dato** | **Condición de selección o repetición** |
| commandParts | String[] |  |
| Actividades generales necesarias para obtener los resultados | Que el arreglo esté inicializado | | |
| Resultado o Postcondición | En caso de que la sintaxis sea válida el método ha de buscar todos los países o todas las ciudades que cumplan con las condiciones impuestas por el usuario | | |
| Salidas | **Nombre salida** | **Tipo de dato** | **Condición de selección o repetición** |
| filterList | String |  |
|  |  |  |

|  |  |  |  |
| --- | --- | --- | --- |
| Nombre o identificador | **R4: Validación de comando ORDENAMIENTO** | | |
| Resumen | El método debe asegurarse de que toda la sintaxis del comando de tipo ordenamiento sea válida | | |
| Entradas | **Nombre entrada** | **Tipo de dato** | **Condición de selección o repetición** |
| commandParts | String[] |  |
| Actividades generales necesarias para obtener los resultados | Que el arreglo esté inicializado | | |
| Resultado o Postcondición | En caso de que la sintaxis sea válida el método ha de buscar todos los países o todas las ciudades que cumplan con las condiciones impuestas por el usuario y organizarlos bajo el criterio que eligió el usuario | | |
| Salidas | **Nombre salida** | **Tipo de dato** | **Condición de selección o repetición** |
| orderedList | String |  |
|  |  |  |

|  |  |  |  |
| --- | --- | --- | --- |
| Nombre o identificador | **R5: Validación de comando DELETE** | | |
| Resumen | El método debe asegurarse de que toda la sintaxis del comando de tipo eliminación sea válida | | |
| Entradas | **Nombre entrada** | **Tipo de dato** | **Condición de selección o repetición** |
| commandParts | String[] |  |
| Actividades generales necesarias para obtener los resultados | Que el arreglo esté inicializado | | |
| Resultado o Postcondición | En caso de que la sintaxis sea válida el método ha de buscar todos los países o todas las ciudades que cumplan con las condiciones impuestas por el usuario borrarlos, en caso de que borre un país las ciudades registradas con su countryCode también han de ser eliminadas. Se enviará un mensaje de confirmación o error respectivamente | | |
| Salidas | **Nombre salida** | **Tipo de dato** | **Condición de selección o repetición** |
| msg (mensaje de confirmación o error) | String |  |
|  |  |  |

|  |  |  |  |
| --- | --- | --- | --- |
| Nombre o identificador | **R6: Importación de datos escritos en formato Json** | | |
| Resumen | El método debe ser capaz de importar todos los datos que estén guardados en un archivo de formato Json al iniciarse el programa | | |
| Entradas | **Nombre entrada** | **Tipo de dato** | **Condición de selección o repetición** |
|  |  |  |
| Actividades generales necesarias para obtener los resultados | Que el archivo de Json exista | | |
| Resultado o Postcondición | Se leerá de coma a coma para crear cada objeto ya sean de tipo country o city con los atributos que están dentro de las llaves para añadirlos a los ArrayList de countries y cities respectivamente. Al final enviará un mensaje al usuario informándole de si la operación fue o no exitosa | | |
| Salidas | **Nombre salida** | **Tipo de dato** | **Condición de selección o repetición** |
| msg (mensaje de confirmación o error) | String |  |
|  |  |  |

|  |  |  |  |
| --- | --- | --- | --- |
| Nombre o identificador | **R7: Guardar datos registrados en formato Json** | | |
| Resumen | El método guardará los registros añadidos en el programa en formato Json para que posteriormente sean recuperados | | |
| Entradas | **Nombre entrada** | **Tipo de dato** | **Condición de selección o repetición** |
|  |  |  |
| Actividades generales necesarias para obtener los resultados | Ninguna | | |
| Resultado o Postcondición | Va a escribir en el archivo todos los registros del programa, y si el archivo no existe lo creará y hará lo mismo. Retornará un mensaje que informará si la operación fue o no exitosa | | |
| Salidas | **Nombre salida** | **Tipo de dato** | **Condición de selección o repetición** |
| msg (mensaje de confirmación o error) | String |  |
|  |  |  |

|  |  |  |  |
| --- | --- | --- | --- |
| Nombre o identificador | **R8: Leer instrucciones en formato SQL** | | |
| Resumen | El método debe ser capaz de leer instrucciones escritas en formato SQL | | |
| Entradas | **Nombre entrada** | **Tipo de dato** | **Condición de selección o repetición** |
| path | String |  |
| Actividades generales necesarias para obtener los resultados | El path debe existir | | |
| Resultado o Postcondición | Va a ejecutar una por una las instrucciones escritas en el archivo de formato SQL. Retornará un mensaje que informará si la operación fue o no exitosa | | |
| Salidas | **Nombre salida** | **Tipo de dato** | **Condición de selección o repetición** |
| msg (mensaje de confirmación o error) | String |  |
|  |  |  |