GraphQL

* Its Stands for **Graph Query Language**.
* GraphQL is a **query language, architecture style**, and **set of tools to create** and **manipulate APIs**.
* GraphQL is **good for large, complex**, and **interrelated data sources.**
* GraphQL has a **Single URL endpoint**.
* GraphQL returns **data in flexible structure** defined by the client.
* GraphQL data is **strongly typed**. So the clients receives data in **predetermined** and **mutually understood formats**.
* With GraphQL, **invalid requests are typically rejected by schema structure**. This results in an **autogenerated**  **error message**.
* In GraphQL call **Rest API** and **another server** also.
* In GraphQL ,deal with three concepts :
  + **Queries** --- get the data from the server .
  + **Mutations** --- Modify the data , update the data on the server (Create , Update , Delete) .
  + **Subscriptions** --- maintain real-time connection with the server.
* **In Queries** ,Describe the data ,

type **Project** {

name : String

tagline : String

contributors : [User]

}

* Ask for what you want ,

{

**Project** (name: "GraphQL") {

tagline

}

}

* Get results ,

{

"**project**": {

"tagline" : "A query language for APIs"

}

}

* In Mutation , “**mutation”**: Indicates that this operation is a mutation.

**mutation** {

**createPost(title: "New Post", content: "This is the content of the new post"**) {

**id**

**title**

**content**

}

}

* In **Subscriptions ,** Indicates that this operation is a mutation.

**subscription** {

**newMessage** {

**id**

**content**

**author** {

**id**

**name**

}

}

}

**EXCEPTION HANDLING**

* Exception is an event that disrupts the normal flow of the program.
* Exception Handling is a mechanism to handle runtime errors

Two types of exceptions

1. Checked Exception

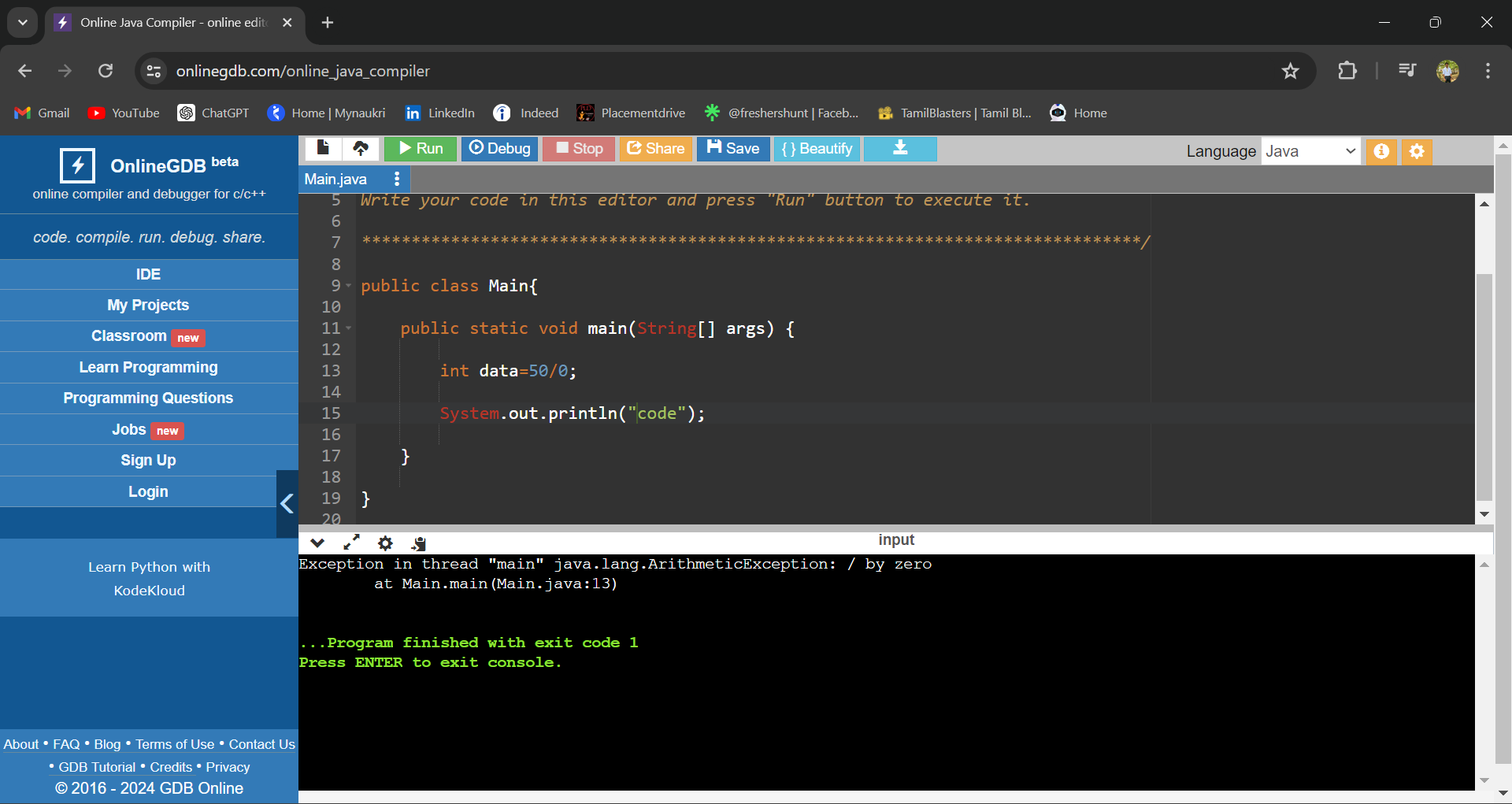
This exception are found during the compile time

1. Unchecked Exception

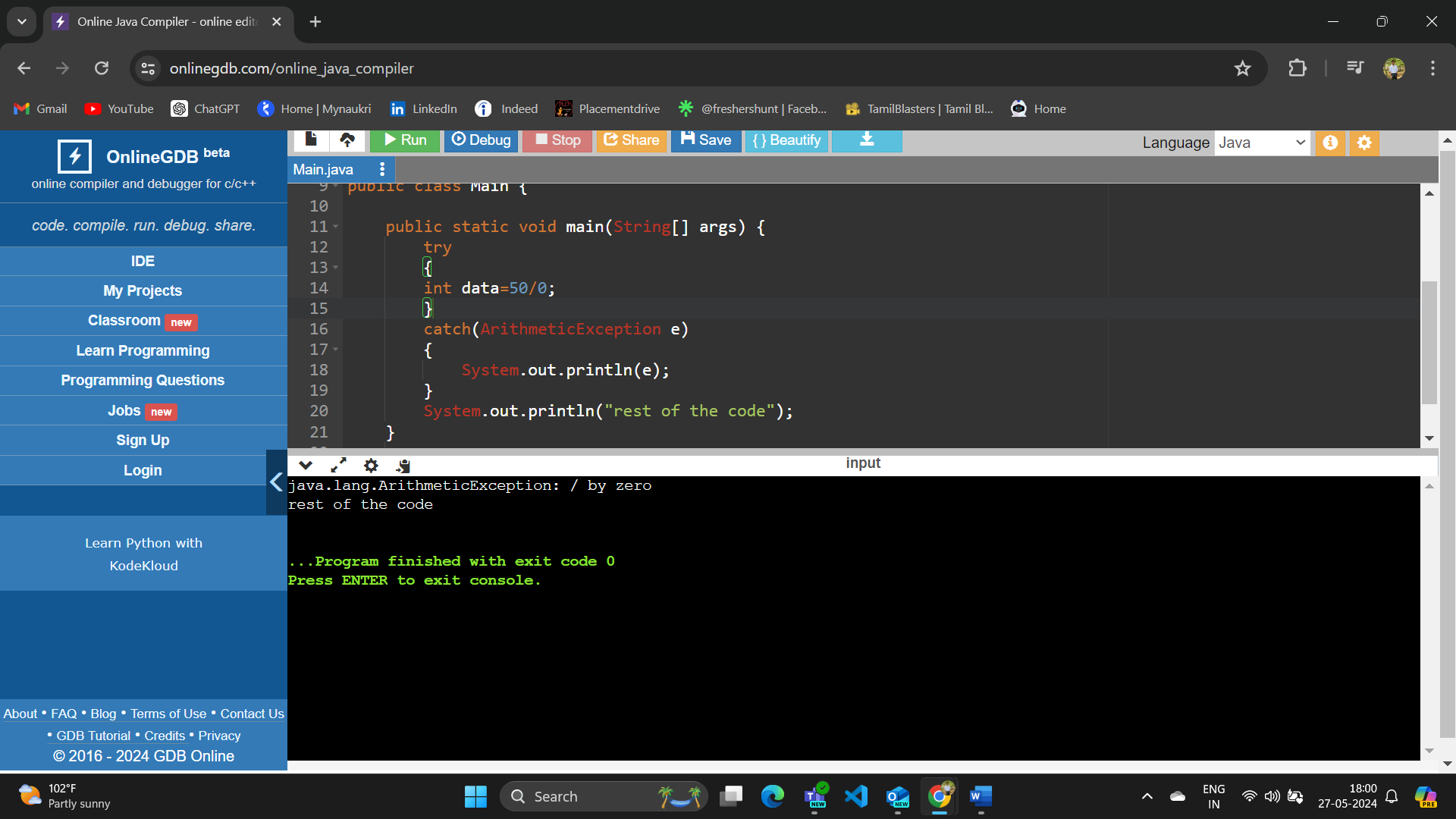
This exception are found during the run time

* Try – This is the block used for checking the exceptions
* Catch -This block is used to catch the Exceptions
* Throw ,Throws – This keyword is used to throw the exception from thetry block to the catch block
* Finally – This block will execute without any interruptions

Consider the code without try-catch block which terminates the program by showing error



Consider this program which does not terminate the flow of the program and also acknowledges the user with the exception catched…



Throw vs Throws

* Java throw keyword is used throw an exception explicitly in the code, inside the function or the block of code.
* throw is used within the method.
* throws is used with the method signature.
* Java throws keyword is used in the method signature to declare an exception which might be thrown by the function while the execution of the code.

JAVA CUSTOM EXCEPTIONS

* In java we can create owr own exceptions based on the user needs..
* Let us consider a program for allowing the persons above and equal to 18 are eligible for voting.

**HTTP STATUS CODE**

**Informational response:**

100 continue: The server has received the request headers and the client should proceed to send the request body.

101 switching protocols: The requester has asked the server to switch protocols and the server has agreed to do so.

103 processing: The server has received and is processing the request, but no response is available yet (WebDAV).

**Successful response:**

200 ok: The request succeeded.

201 created: The request fulfilled, creation of a new complete.

202 accepted: The request accepted but processing is not complete.

204 no content: The server successfully processed the request, but no content.

205 reset content: The server successfully processed the request, but is not returning any content and requires that the requester reset the document view.

Redirection Message:

300 multiple choices: Indicates multiple options for the resource from which the client may choose.

301 moved permanently: request has been changed permanently,the new url is given.

302 found: request has been changed temporarily,the new url is given.

304 not modified: the resource has not been modified since the version specified by the request headers.

**Client Error Responses**:

400 Bad Request: The server cannot or will not process the request due to a client error

401 unauthorized: The request requires user authentication.

402 payment required: This status code is reserved for future use.

403 forbidden: The server understood the request, but refuses to authorize it.

404 not found: The server cannot find the requested resource.

415 unsupported media type: media type which the server or resource does not support.

**Server Error Responses**:

500 internal server error: A generic error message, given when an unexpected condition was encountered.

502 bad gateway: The server, while acting as a gateway or proxy, received an invalid response from the upstream server.

503 service unavailable: The server is currently unable to handle the request due to temporary overload or scheduled maintenance.