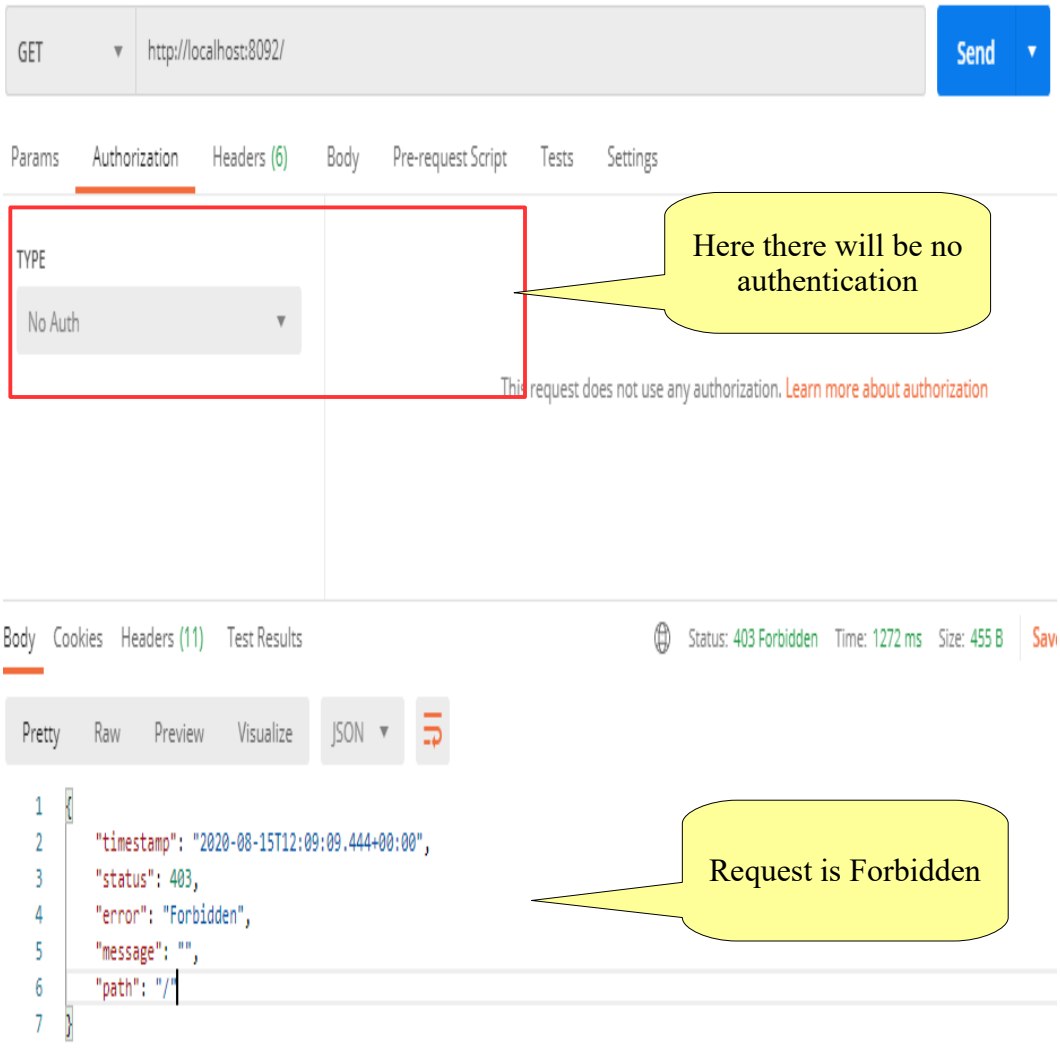
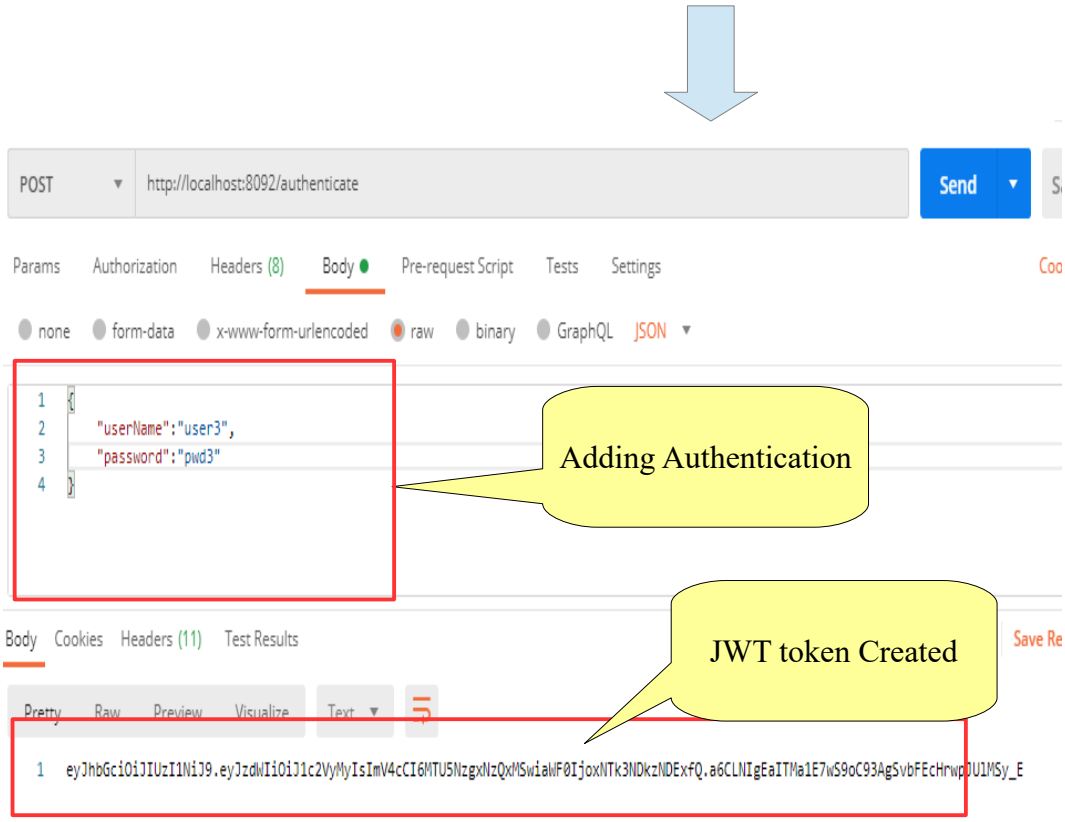
AWT authentication

1.Checking Get Request :

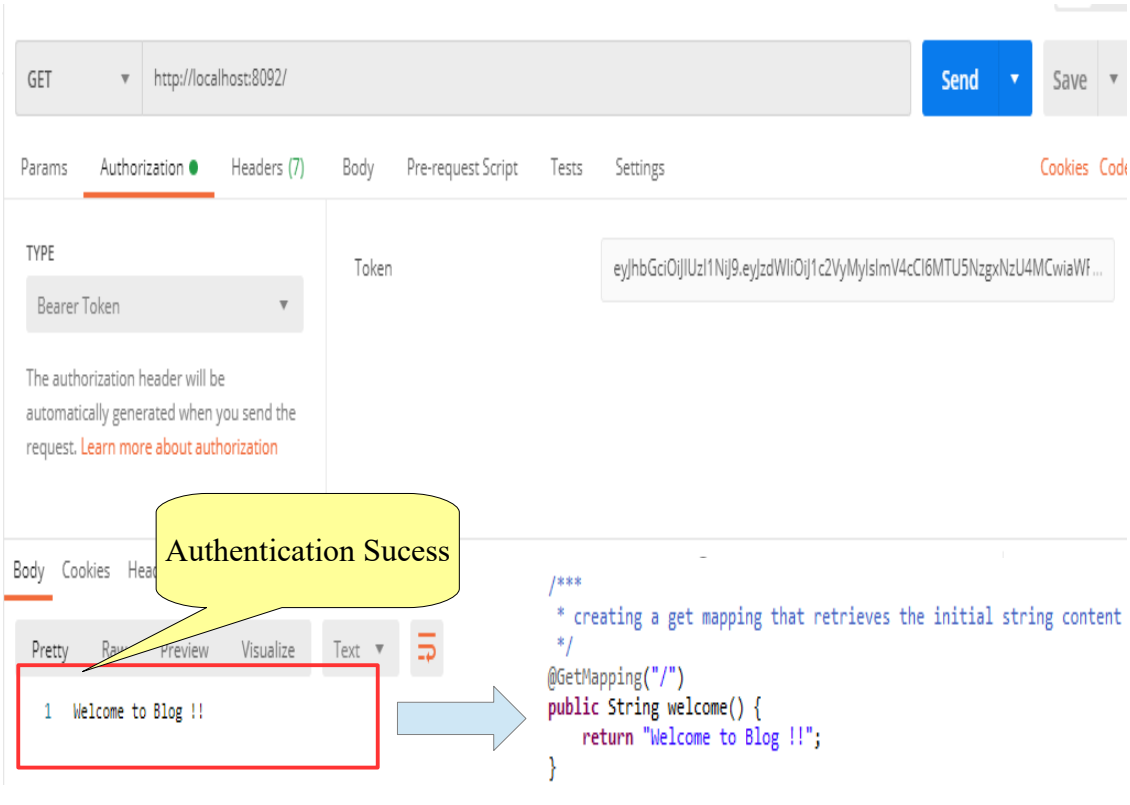


2.Generating JWT token

```
/**
 * generateToken method is used to generate the token
 *
 * @param authRequest
 * @return Strig
 * @throws Exception
 */
@PostMapping("/authenticate")
public String generateToken(@RequestBody AuthRequest authRequest) throws Exception {
    try {
        authenticationManager.authenticate(
                new UsernamePasswordAuthenticationToken(authRequest.getUserName(), authRequest
    } catch (Exception ex) {
        throw new Exception("inavalid username/password");
    }
    return jwtUtil.generateToken(authRequest.getUserName());
}
```



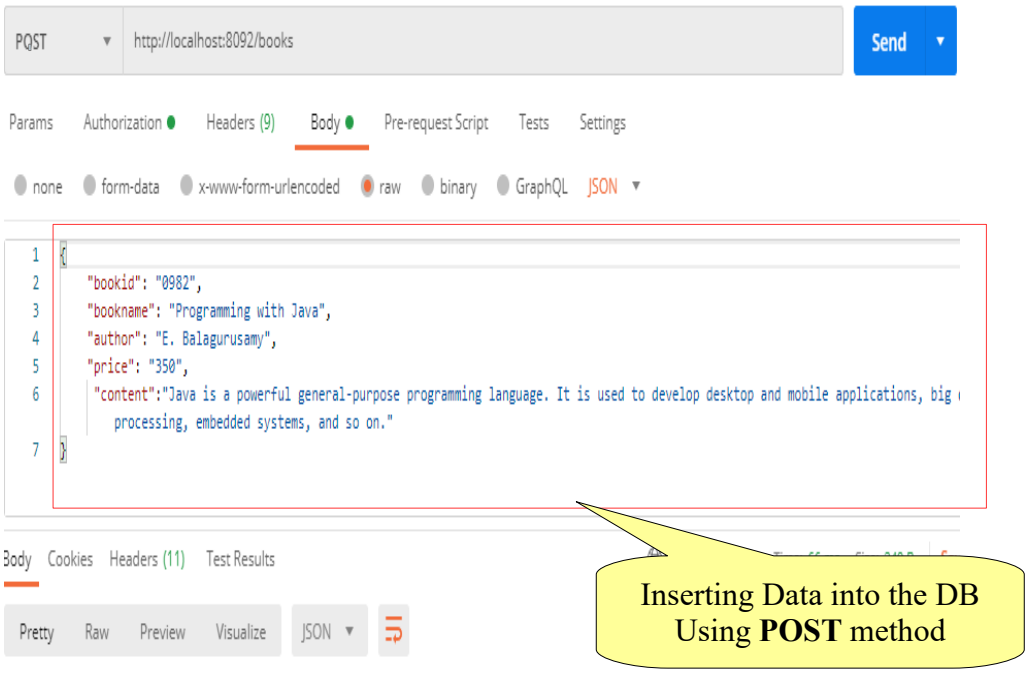
3.Accessing API with JWT token



Performing CRUD operations:

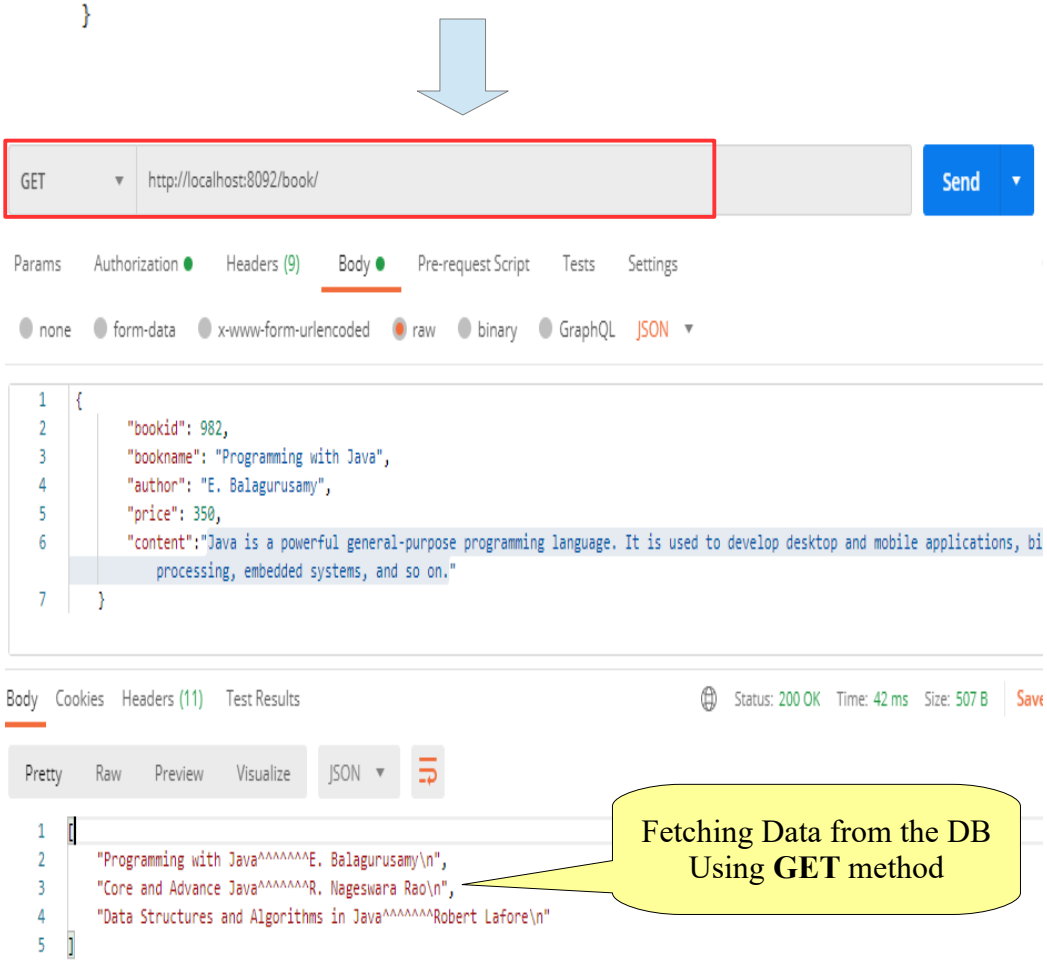
Inserting Data into DataBase :

```java
/**
 * creating post mapping that post the book detail in the database
 *
 * @param books
 * @return int
 */
@PostMapping("/books")
private int saveBook(@RequestBody Books books) {
    booksService.saveOrUpdate(books);
    return books.getBookid();
}
```



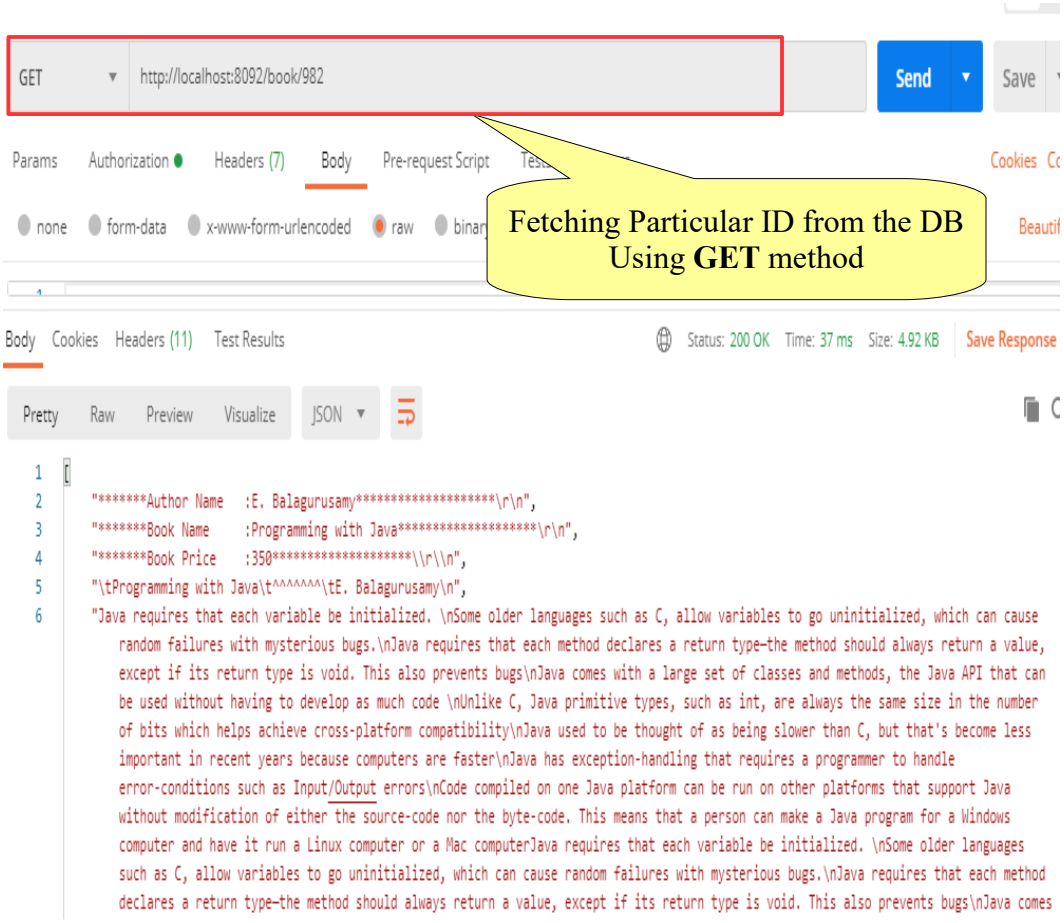
Getting All Data From Database:

```java
/**
 * creating a get mapping that retrieves all the books detail from the database
 *
 * @return List<String>
 */
@GetMapping("/book")
private List<String> getAllBooks() {
    List<Books> booksList = booksService.getAllBooks();
      List<String> bookDetails = booksList.stream().map(book ->{
         StringBuffer ff = new StringBuffer();
         String bookDetail = book.getBookname()+"^^^^^^^"+book.getAuthor()+"\n";
         return bookDetail;
    }).collect(Collectors.toList());
    return bookDetails;
}
```



Fetching Particular ID from DataBase :

```
/**
 * creating a get mapping that retrieves the detail of a specific book
 *
 * @param bookid
 * @return List<String>
 */
@GetMapping("/book/{bookid}")
private List<String> getBooks(@PathVariable("bookid") int bookid) {
    List<String> stringList = new ArrayList<>();
    Books book = booksService.getBooksById(bookid);
    StringBuilder ff = new StringBuilder();
     String bookDetail = "\t"+book.getBookname()+"\t"+"^^^^^^^"+"\t"+book.getAuthor();
     stringList.add("*******Author Name   :"+book.getAuthor()+"********************"+"\r\n");
     stringList.add("*******Book Name     :"+book.getBookname()+"********************"+"\r\n");
     stringList.add("*******Book Price    :"+book.getPrice()+"********************"+"\\r\\n");
     stringList.add(bookDetail+"\n");
     stringList.add(Const.TEXT_CONTENT);
     stringList.add(Const.TEXT_CONTENT+'\n');
    return stringList;
}
```


GET
http://localhost:8092/book/982
Send
Save
Params
Authorization
Headers (7)
Body
Pre-request Script
Cookies
none
form-data
x-www-form-urlencoded
raw
Fetching Particular ID from the DB
Using GET method
Body
Cookies
Headers (11)
Test Results
Status: 200 OK
Time: 37 ms
Size: 4.92 KB
Save Response
Pretty
Raw
Preview
Visualize
JSON
1 [
2 "*******Author Name :E. Balagurusamy**********\r\n",
3 "*******Book Name :Programming with Java**********\r\n",
4 "*******Book Price :350**********\\r\\n",
5 "\tProgramming with Java\t^^^^^^^\tE. Balagurusamy\n",
6 "Java requires that each variable be initialized. \nSome older languages such as C, allow variables to go uninitialized, which can cause random failures with mysterious bugs.\nJava requires that each method declares a return type—the method should always return a value, except if its return type is void. This also prevents bugs\nJava comes with a large set of classes and methods, the Java API that can be used without having to develop as much code \nUnlike C, Java primitive types, such as int, are always the same size in the number of bits which helps achieve cross-platform compatibility\nJava used to be thought of as being slower than C, but that's become less important in recent years because computers are faster\nJava has exception-handling that requires a programmer to handle error-conditions such as Input/Output errors\nCode compiled on one Java platform can be run on other platforms that support Java without modification of either the source-code nor the byte-code. This means that a person can make a Java program for a Windows computer and have it run a Linux computer or a Mac computerJava requires that each variable be initialized. \nSome older languages such as C, allow variables to go uninitialized, which can cause random failures with mysterious bugs.\nJava requires that each method declares a return type—the method should always return a value, except if its return type is void. This also prevents bugs\nJava comes

THANK YOU !!!