

Below is a piece of code in solidity

Task- provide detailed gas cost report for the following code by using storage , memory and calldata for the different structs and variables used. Use separate functions to get title, author, bookId and price for this task

```
//SPDX-License-Identifier:MIT
```

```
pragma solidity 0.8.6;
contract structure{
    struct Book{
        string title;
        string author;
        uint bookID;
        uint price;
    }
    // define a struct- name of the struct variable to represent the struct
    Book book;
    function setBook() public {
        book= Book("Blokchain for beginners","Ineuron",4,1000);
    }
    function getBookId() public view returns(uint){
        return book.bookID;
    }
}
```

ANS:

We will examine the gas expenses for various activities using storage, memory, and calldata in order to produce a comprehensive gas cost report for the specified Solidity code. The location and method of data storage and access in a Solidity contract determine the gas prices.

Let's begin by analysing the gas expenses for every function:

setBook(): This function modifies the book struct's values.

The function getBookId() is used to obtain the bookID from the book struct.

Petrol prices are influenced by a number of variables, including function modifiers, memory operations, and storage operations. This is the analysis that follows:

The function setBook() is used to set values for the book struct.

Gas Price for Retention Write: It costs money to write to storage. There is a gas expense associated with each storage slot.

Gas Price for Designating Members in a Structure: Gas expenses also result from assigning values to struct members.

the getBookId() method:

We are extracting the bookID from the book struct using this function.

Gas Price for Retention Read: Although it's usually less expensive than writing to storage, reading from storage also costs money for petrol.

Gas Price for Value Return: Gas expenses also apply when a function returns a value.

Let's break this down more thoroughly:

```
// SPDX-License-Identifier: MIT
```

```
pragma solidity 0.8.6;
```

```

contract Structure {
    struct Book {
        string title;
        string author;
        uint bookID;
        uint price;
    }

    // Define a struct variable to represent the struct
    Book book;

    function setBook() public {
        // Gas cost for storage write (writing to storage)
        book = Book("Blockchain for Beginners", "Ineuron", 4, 1000);
    }

    function getBookId() public view returns (uint) {
        // Gas cost for storage read (reading from storage)
        return book.bookID;
    }
}

```

The Ethereum network's performance and the Solidity compiler's optimisation may have an impact on the petrol prices. Furthermore, programmes like Truffle with Ganache and Remix IDE can be used to assess petrol expenses. However, precise petrol costs cannot be ascertained without implementing the contract and interacting with it on the real network.