MageINIC

Price Per Customer GraphQl

- The Price per customer module offer graphql to get custom price as per product and customer.

How to install the extension

1. Upload via ZIP/Archive

- Back up your store's database and web directory
- Download the extension zip file and extract the zip file
- Transfer the **app** folder into the **Magento2 root** directory on the server
- Enable maintenance mode **php bin/magento maintenance:enable**
- Run the command **php bin/magento module:enable MageINIC_Core MageINIC_PricePerCustomerGraphQl**
- Run the command **php bin/magento setup:upgrade** to install the extension
- Run the command **php bin/magento cache:clean** to clean the cache
- Disable maintenance mode **php bin/magento maintenance:disable**
- Check the store environment if everything is working properly. If, at any point, the store displays an error, check the Magento logger for more details.

Store Configuration

- Please include the following query in the body section to fetch the module store configuration values.

  End Point : {{baseurl}}/graphql

```
{
   storeConfig{
      pricepercustomer_enable,
      customer_price_link_name,
      customer_price_page_title,
      customer_price_show_link_in_top,
      customer_price_Show_link_footer,
      customer_price_page_layout
   }
}
```

Response :

```
{
    "data": {
        "storeConfig": {
            "pricepercustomer_enable": 1,
            "customer_price_link_name": "just for you! \"Offers\"",
            "customer_price_page_title": "Special Price Products",
            "customer_price_show_link_in_top": 1,
            "customer_price_Show_link_footer": 1,
            "customer_price_page_layout": "1column"
        }
    }
}
```

You receive store configuration data set on the backend, including:

- **pricepercustomer_enable**: A module configuration that enable or disable the module functionality.
- **customer_price_link_name** : You get custom price link text in it.
- **customer_price_page_title** : You receive custom page title wich set on backend.
- **customer_price_show_link_in_top** : Receive value 1 when it set yes value for top link other wise value is 0.
- **customer_price_Show_link_footer** : Receive value 1 when it set yes value for footer link other wise value is 0.
- **customer_price_page_layout** : You receive page layout on this.

Price Per Customer Filter

End Point : {{baseurl}}/graphql
Request : POST
Note : You can filter record as per you want using below options like:
entity_id,customer_id,product_id.

```
{
 PricePerCustomer(filter: {
   entity_id: {eq: "1"},
   customer_id: {eq: "2"},
   product_id: {eq: "1"}
 }, pageSize: 6, currentPage: 1) {
```

```
    pricePerCustomerList {
      customer_id
      price
      product_id
      quantity
    }
    total_count
    total_pages
  }
}
```

Response :

```
{
    "data": {
      "pricePerCustomer": {
        "pricePerCustomerList": [
          {
            "customer_id": 2,
            "price": "10",
            "product_id": 1,
            "quantity": 2
          }
        ],
        "total_count": 1,
        "total_pages": 1
      }
    }
}
```

User Custom Product Price

End Point : {{baseurl}}/graphql
Request: POST
Note : You need frontend customer token for run query and pass particular product sku on it.

```
{
  products(filter: { sku: { eq: "24-MB01" }}) {
    total_count
    items {
      price_range {
        minimum_price {
          final_price {
```

```
          value
          currency
        }
      }
    maximum_price {

     final_price {
       value
       currency
      }
     }
    }

   }
  }
}
```

Response :

```
{
  "data": {
    "products": {
      "total_count": 1,
      "items": [
        {
          "price_range": {
            "minimum_price": {
              "final_price": {
                "value": 20,
                "currency": "USD"
              }
            },
            "maximum_price": {
              "final_price": {
                "value": 20,
                "currency": "USD"
              }
            }
          }
        }
      ]
    }
  }
}
```