

# Shopify - GraphQL Queries

Generated: 19/11/2025, 18:42:10

Description: Shopify is a cloud-based e-commerce platform that allows individuals and businesses to create and manage their own online stores.

Total Queries: 14

## 1. Get All Products

```
{  
  "query": "query { products(first: 250) { edges { node { id title handle  
    descriptionHtml vendor productType status tags createdAt updatedAt variants(first:  
    10) { edges { node { id sku title price inventoryQuantity compareAtPrice } } } }  
    images(first: 10) { edges { node { id url altText width height } } } } pageInfo {  
    hasNextPage hasPreviousPage endCursor } } }"  
}
```

## 2. Create product

```
{  
  "query": "mutation CreateProduct($product: ProductCreateInput!, $media:  
[CreateMediaInput!]) { productCreate(product: $product, media: $media) { product {  
    id title handle status vendor productType descriptionHtml tags options { id name  
    position optionValues { id name hasVariants } } seo { title description }  
    media(first: 10) { edges { node { mediaContentType alt } } } variants(first: 10) {  
      edges { node { id title sku price inventoryItem { id tracked } } } }  
      metafields(first: 20) { edges { node { id namespace key type value } } } }  
      userErrors { field message } } }",  
  "variables": {  
    "product": {  
      "title": "FaizKahn-Test",  
      "descriptionHtml": "<p>Premium cotton socks</p>",  
      "vendor": "Sock Co",  
      "productType": "Socks",  
      "tags": ["cotton", "unisex", "premium"],  
      "status": "ACTIVE",  
      "handle": "faizsse-socks2",  
      "seo": { "title": "Cool Socks", "description": "Comfortable, breathable  
socks." },  
      "productOptions": [  
        { "name": "Color", "values": [ { "name": "Red" }, { "name": "Blue" } ] },  
        { "name": "Size", "values": [ { "name": "Small" }, { "name": "Large" } ] }  
      ]  
    }  
  }  
}
```

```
        ] ,  
  
        "metafields": [  
  
            { "namespace": "custom", "key": "color", "type": "single_line_text_field",  
              "value": "Brown" } ,  
  
            { "namespace": "custom", "key": "category", "type":  
              "single_line_text_field", "value": "Sofa" } ,  
  
            { "namespace": "custom", "key": "collection", "type":  
              "single_line_text_field", "value": "Adia" } ,  
  
            { "namespace": "custom", "key": "style", "type": "single_line_text_field",  
              "value": "Modern" } ,  
  
            { "namespace": "custom", "key": "material", "type":  
              "single_line_text_field", "value": "Mesh and Steel" } ,  
  
            { "namespace": "custom", "key": "product_dimension", "type":  
              "single_line_text_field", "value": "24x30x20" } ,  
  
            { "namespace": "custom", "key": "short_description", "type":  
              "single_line_text_field", "value": "Compact ergonomic chair." } ,  
  
            { "namespace": "custom", "key": "long_description", "type":  
              "multi_line_text_field", "value": "Extended marketing copy..." }  
        ]  
    } ,  
  
    "media": [  
  
        {  
  
            "originalSource": "https://shopify-staged-uploads.storage.googleapis.com/tm  
p/56620515381/files/9d9f9a78-0bb3-4c17-9912-5a152ae1717e/aiesta-download-17612194  
69778.png" ,  
  
            "alt": "Red socks front view" ,  
  
            "mediaContentType": "IMAGE"  
        }  
    ]  
}
```

### 3. Delete product:

```
{  
  
  "query": "mutation productDelete($input: ProductDeleteInput!) {  
    productDelete(input: $input) { deletedProductId userErrors { field message } } }",  
  
  "variables": {  
  
    "input": {  
  
      "id": "gid://shopify/Product/id"  
  
    }  
  
  }  
  
}
```

### 4. UpdateProduct

```
{  
  
  "query": "mutation UpdateProduct($input: ProductInput!) { productUpdate(input:  
    $input) { product { id title handle status vendor productType descriptionHtml tags  
      seo { title description } options { id name position optionValues { id name } } }  
      metafields(first: 50) { edges { node { id namespace key type value } } } }  
      userErrors { field message } } }",  
  
  "variables": {  
  
    "input": {  
  
      "id": "gid://shopify/Product/8736248917364",  
  
      "title": "Updated Product Title via GraphQL",  
  
      "descriptionHtml": "<p>Updated description from Postman!</p>",  
  
      "status": "ACTIVE",  
  
      "vendor": "Sock Co",  
  
      "productType": "Socks",  
  
      "tags": [ "cotton", "unisex", "premium" ],  
  
      "seo": {  
  
        "title": "Cool Socks - Updated",  
  
        "description": "Comfortable, breathable socks (updated)."  
      },  
  
    },  
  
  }
```

```

    "options": [
      { "id": "gid://shopify/ProductOption/123", "name": "Color" },
      { "name": "Size", "optionValues": [ { "name": "Small" }, { "name": "Large" } ] }
    ],
    "metafields": [
      { "id": "gid://shopify/Metafield/111", "value": "Brown" },
      {
        "namespace": "custom",
        "key": "material",
        "type": "single_line_text_field",
        "value": "Mesh and Steel"
      }
    ]
  }
}

```

## 5. Get All Orders

```
{
  "query": "query {\n  orders(first: 250, sortKey: PROCESSED_AT, reverse: true) {\n    edges {\n      cursor\n      node {\n        id\n        name\n        createdAt\n        processedAt\n        displayFinancialStatus\n        displayFulfillmentStatus\n        closedAt\n        currencyCode\n        currentSubtotalPriceSet {\n          shopMoney {\n            amount\n            currencyCode\n          }\n        }\n        currentTotalPriceSet {\n          shopMoney {\n            amount\n            currencyCode\n          }\n        }\n        currentTotalTaxSet {\n          shopMoney {\n            amount\n            currencyCode\n          }\n        }\n        totalWeight\n        customer {\n          id\n          firstName\n          lastName\n          email\n          phone\n          shippingAddress {\n            name\n            address1\n            address2\n            city\n            province\n            country\n            zip\n            phone\n            address1\n            address2\n            city\n            province\n            country\n            zip\n            trackingInfo {\n              number\n              company\n              url\n            }\n            status\n            createdAt\n          }\n        }\n        fulfillments(first: 10)\n        lineItems(first: 50) {\n          edges {\n            node {\n              id\n
```

```

title\n           sku\n           quantity\n
originalUnitPriceSet {\n                     shopMoney {\n                           amount\n
                           currencyCode\n
                           }\n
discountedTotalSet {\n                     shopMoney {\n                           amount\n
                           currencyCode\n
                           }\n
                           }\n
                           id\n           title\n           }\n
variant {\n           id\n           title\n           }\n
}\n
}\n
}\n
hasPreviousPage\n      startCursor\n      endCursor\n      }\n
}\n
}

```

## 6. Get ALL Inventory

```
{"query": "{ products(first: 10) { edges { node { id title variants(first: 10) {
edges { node { id title sku inventoryItem { id inventoryLevels(first: 5) { edges {
node { location { name } quantities(names: \\\"available\\\") { name quantity } }
} } } } } } } } } }"}}
```

## 7. Update that variant (SKU, price)

```
{
  "query": "mutation productVariantsBulkUpdate($productId: ID!, $variants: [ProductVariantsBulkInput!]!) { productVariantsBulkUpdate(productId: $productId, variants: $variants) { productVariants { id price inventoryItem { id sku } } userErrors { field message } } }",
  "variables": {
    "productId": "gid://shopify/Product/7996623487029",
    "variants": [
      {
        "id": "gid://shopify/ProductVariant/44796944154677",
        "price": "299.99",
        "inventoryItem": {
          "sku": "FURN-001"
        }
      }
    ]
  }
}
```

## **8. Add inventory to a product(quantity...)**

```
{  
  "query": "mutation { inventorySetQuantities(input: { name: \"available\", reason:  
    \"correction\", ignoreCompareQuantity: true, quantities: [{ inventoryItemId:  
      \"gid://shopify/InventoryItem/44182182953013\", locationId:  
      \"gid://shopify/Location/62354096181\", quantity: 999 }] }) { userErrors { field  
      message } } }"  
}
```

## **9. Fetch location id of product using Title:**

```
{  
  "query": "query ProductLocationIds($query: String!, $firstProducts: Int = 20,  
  $firstVariants: Int = 50, $firstLevels: Int = 50) { products(first: $firstProducts,  
  query: $query) { edges { node { id title variants(first: $firstVariants) { edges {  
      node { id inventoryItem { id inventoryLevels(first: $firstLevels) { edges { node {  
          location { id name } } } pageInfo { hasNextPage endCursor } } } } pageInfo {  
          hasNextPage endCursor } } } pageInfo { hasNextPage endCursor } } }  
  \"variables\": {  
    \"query\": \"title:*socks*\",  
    \"firstProducts\": 10,  
    \"firstVariants\": 50,  
    \"firstLevels\": 50  
  }  
}
```

## **10. Change image file into url**

```
{  
  "query": "mutation stagedUploadsCreate($input: [StagedUploadInput!]!) {  
    stagedUploadsCreate(input: $input) { stagedTargets { resourceUrl url parameters {  
        name value } } userErrors { field message } } }  
  \"variables\": {  
    \"input\": [  
      {  
        \"resource\": \"IMAGE\",  
        \"filename\": \"\",  
        \"mimeType\": \"image/png\",  
        \"fileSize\": \"245678\"  
      }  
    ]  
  }
```

```
        ]  
    }  
}
```

## 11. Update Existing Inventory

```
{  
  "query": "mutation { inventorySetQuantities(input: { name: \"available\", reason:  
\"correction\", ignoreCompareQuantity: true, quantities: [{ inventoryItemId:  
\"gid://shopify/InventoryItem/46896607035445\", locationId:  
\"gid://shopify/Location/62354096181\", quantity: 100 }] }) { userErrors { field  
message } } }"  
}
```

## 12. Get\_Inventory\_Location\_ID

```
{  
  "query": "query GetLocations($first: Int = 50) { locations(first: $first) { edges  
{ node { id name legacyResourceId address { address1 address2 city province country  
countryCode zip } } } pageInfo { hasNextPage endCursor } } }",  
  "variables": { "first": 20 }  
}
```

## 13. Generate\_Id\_For\_Image

```
{ "query": "mutation fileCreate($files: [FileCreateInput!]!) { fileCreate(files:  
$files) { files { ... on MediaImage { id fileStatus image { url altText width  
height } } userErrors { field message } } }", "variables": { "files": [ { "alt":  
"Test image", "contentType": "IMAGE", "originalSource": "" } ] } }
```

## 14. Get\_CDN\_Image\_Url

```
{ "query": "query($id: ID!) { node(id: $id) { ... on MediaImage { id fileStatus  
image { url altText width height } } } }", "variables": { "id": "" } }
```