**Task:-** Create a customer attribute called "Last ordered date", when ever customer placed an order that custom attribute date should be updated. This attribute should be displayed on the admin panel.  
Should use event & observers concept.

**Create a customer attribute called "Last ordered date" :-**

**Using install data to create a custom attribute**

**I95dev / Customerattribute /Setup/ InstallData.php**

**<?php**

**namespace I95dev\Customerattribute\Setup;**

**use Magento\Customer\Setup\CustomerSetupFactory;**

**use Magento\Customer\Model\Customer;**

**use Magento\Eav\Model\Entity\Attribute\Set as AttributeSet;**

**use Magento\Eav\Model\Entity\Attribute\SetFactory as AttributeSetFactory;**

**use Magento\Framework\Setup\InstallDataInterface;**

**use Magento\Framework\Setup\ModuleContextInterface;**

**use Magento\Framework\Setup\ModuleDataSetupInterface;**

**/\*\***

**\* @codeCoverageIgnore**

**\*/**

**class InstallData implements InstallDataInterface**

**{**

**/\*\***

**\* @var CustomerSetupFactory**

**\*/**

**protected $customerSetupFactory;**

**/\*\***

**\* @var AttributeSetFactory**

**\*/**

**protected $attributeSetFactory;**

**/\*\***

**\* @param CustomerSetupFactory $customerSetupFactory**

**\* @param AttributeSetFactory $attributeSetFactory**

**\*/**

**public function \_\_construct(**

**CustomerSetupFactory $customerSetupFactory,**

**AttributeSetFactory $attributeSetFactory**

**) {**

**$this->customerSetupFactory = $customerSetupFactory;**

**$this->attributeSetFactory = $attributeSetFactory;**

**}**

**/\*\***

**\* {@inheritdoc}**

**\*/**

**public function install(ModuleDataSetupInterface $setup, ModuleContextInterface $context)**

**{**

**/\*\* @var CustomerSetup $customerSetup \*/**

**$customerSetup = $this->customerSetupFactory->create(['setup' => $setup]);**

**$customerEntity = $customerSetup->getEavConfig()->getEntityType('customer');**

**$attributeSetId = $customerEntity->getDefaultAttributeSetId();**

**/\*\* @var $attributeSet AttributeSet \*/**

**$attributeSet = $this->attributeSetFactory->create();**

**$attributeGroupId = $attributeSet->getDefaultGroupId($attributeSetId);**

**// The Last\_ordered\_date is the attribute code**

**// We add this attribute before the Email field so I set it to 79**

**$customerSetup->addAttribute(Customer::ENTITY, 'Last\_ordered\_date', [**

**'type' => 'datetime', // The types are datetime, decimal, int, text, varchar**

**'label' => 'Last Ordered Date', // it show grid column**

**'input' => 'date', // input type**

**'required' => false, // this field is required**

**'visible' => true, // this attribute show visible or not**

**'user\_defined' => true,**

**'sort\_order' => 79, // The sort order will be position this attribute will be displayed**

**'position' => 79,**

**'system' => 0,**

**'is\_used\_in\_grid' => true, //** Ready for use in the customer grid

**'is\_visible\_in\_grid' => true,//** Visible in the customer grid

**'is\_html\_allowed\_on\_front' => false,**

**'visible\_on\_front' => true// visible in store front**

**]);**

**$attribute = $customerSetup->getEavConfig()->getAttribute(Customer::ENTITY, 'Last\_ordered\_date')**

**->addData([**

**'attribute\_set\_id' => $attributeSetId,**

**'attribute\_group\_id' => $attributeGroupId,**

**'used\_in\_forms' => ['adminhtml\_customer', 'customer\_account\_edit'],**

**]);**

**$attribute->save();**

**}**

**}**

**when ever customer placed an order that custom attribute date should be updated:-**

**I95dev/ Customerattribute /etc/events.xml**

**<?xml version="1.0"?>**

**<config xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"**

**xsi:noNamespaceSchemaLocation="urn:magento:framework:Event/etc/events.xsd">**

**<event name="sales\_order\_place\_after">**

**<observer name="place\_order\_after" instance="I95dev\Customerattribute\Observer\Orderplaceafter"/>**

**</event>**

**</config>**

**Observer:- I95dev\Customerattribute\Observer\Orderplaceafter**

**<?php**

**namespace I95Dev\ Customerattribute \Observer;**

**use Magento\Framework\Event\ObserverInterface;**

**class Orderplaceafter implements ObserverInterface**

**{**

**protected $customerRepository;**

**public function \_\_construct(**

**\Magento\Customer\Api\CustomerRepositoryInterfaceFactory $customerRepositoryFactory ) {**

**$this->customerRepository = $customerRepositoryFactory->create();**

**}**

**public function execute(\Magento\Framework\Event\Observer $observer)**

**{**

**$orderdata = $observer->getEvent()->getOrder();**

**$customerdata = $this->customerRepository->getById($ orderdata ->getData('customer\_id'));**

**$value= ‘**09/3/2022’;

**$customerdata ->setCustomAttribute('last\_order\_date', $value);**

**$this->customerRepository->save($customerdata);**

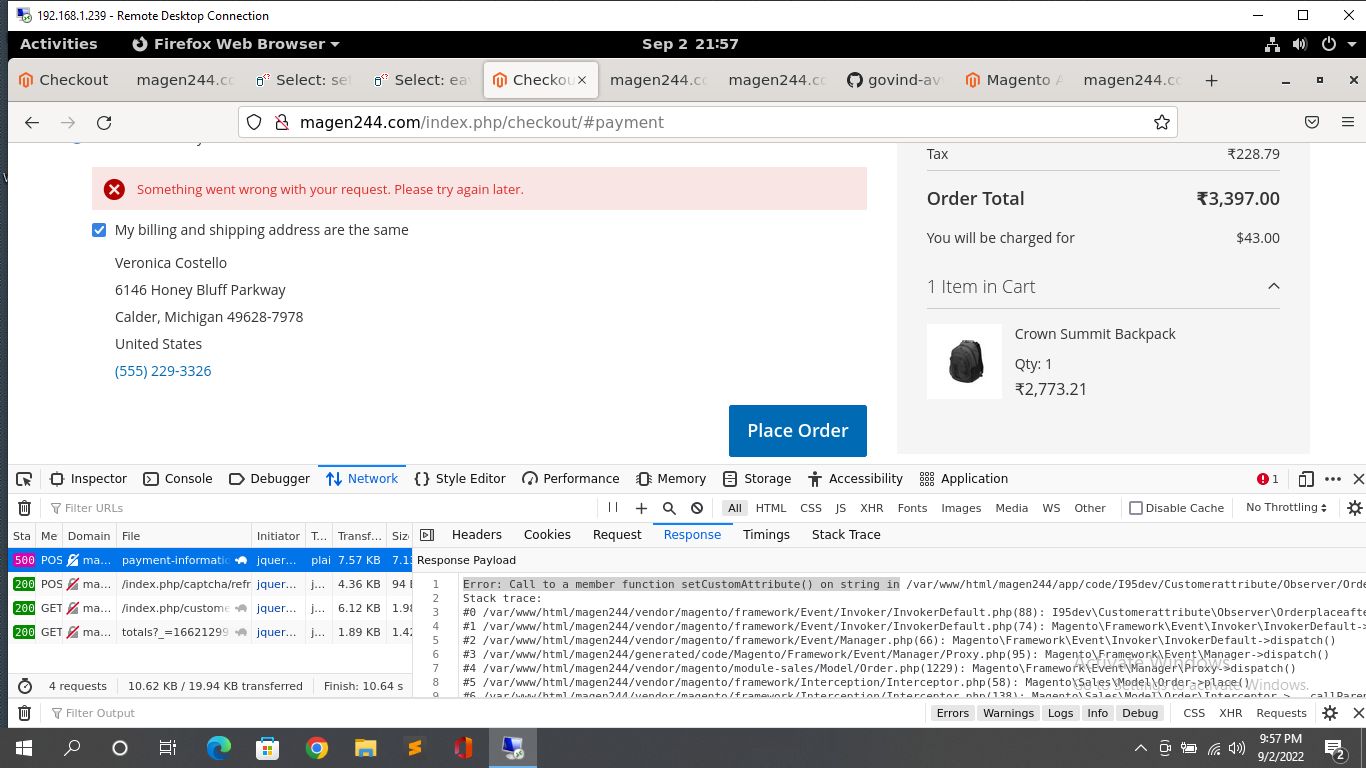
**}**

**}**

**How we know event was triggered:-**

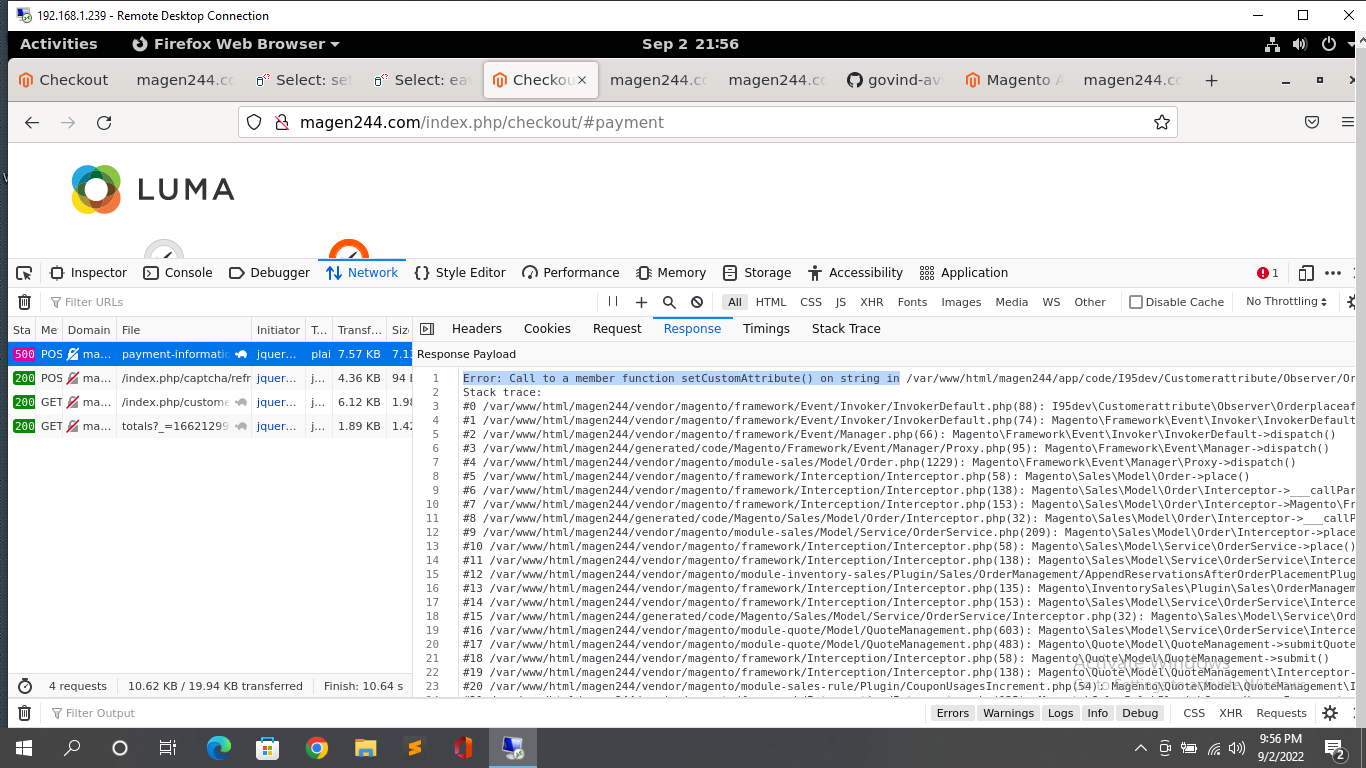
**When we are placing order in that time we have to open inspect in browser**

**Goto->network->click on payment info then click on Response here we get an any error .in this reponse tab it will show.**



**In the response we are getting error lick function setcustomattribute on string**

**It means event was triggered but error in observer file.**



The output of task:-

