**Pra-pmt-lambda-apps-service**

**Request Flow Diagram: -**

**Request input**

**public class Event {**

**private String lob;**

**private String portfolioType;**

**private String uploadTo;**

**private String startDate;**

**private String endDate;**

**private boolean skipUploadToVendor;**

**private String sid;**

**private String actionType;**

**private String jiraTicket;**

**private String comments;**

**private List<String> portfolioId;**

**}**

**@Controller**

**@RequiredArgsConstructor (onConstructor\_ = @Autowired)**

**public class HoldingsUploadController {**

**@Autowired**

**private Handler handler;**

**@Autowired**

**private Audit audit;**

**@PostMapping("/uploadHoldings")**

**public ResponseEntity<Boolean> uploadHoldings (@RequestBody Event event) throws JsonProcessingException {**

**ObjectMapper objectMapper = new ObjectMapper();**

**handler.accept(objectMapper.writeValueAsString(event));**

**return new ResponseEntity<>(HttpStatus.OK);**

**}**

**}**

**bean-context.xml**

**<?xml version="1.0" encoding="UTF-8"?>**

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

**xmlns:util="http://www.springframework.org/schema/util xmlns="http://www.springframework.org/schema/beans\***

**xsi:schemaLocation="http://www.springframework.org/schema/beans http://www.springframework.org/schema/beans/spring-beans.xsd http://www.springframework.org/schema/util http://www.springframework.org/schema/util/spring-util.xsd">**

**<description>**

**This has settings for the Lob rules.**

**</description>**

**<util:map id="uploadGatewayMap key-type="java.lang.String" value-type="com.jpmorgan.wm.pra.domain.port.HoldingsUploader">**

**<entry key="EFG\_HYPO\_MSTAR" value-ref="efgHypoMStarEventHandler" />**

**<entry key="EFG\_GRID\_BBG\* value-ref="efgGridBbgEventHandler"/>**

**<entry key="EFG\_PAR\_BBG\* value-ref="efgHypoBbgEventHandler"/>**

**<entry key="ESG\_PAR\_MSTAR\* value-ref="esgHypoMStarEventHandler"/>**

**</util:map>**

**</beans>**

**@Component**

**@slf4j**

**public class Handler implements Consumer<String> {**

**@Resource**

**private Map<String, HoldingsUploader> uploadGatewayMap;**

**private HoldingsUploader holdingsUploader;**

**@Override**

**public void accept(final String input) {**

**log.info("Received: {}", input);**

**ObjectMapper mapper = new ObjectMapper();**

**try {**

**Event event = mapper.readValue(input, Event.class);**

**String uploadType = Constants.UPLOAD\_TYPE.formatted (event.getLob(), event.getPortfolioType(), event.getUp log.info("Event {}", event);**

**log.info("UploadType {}", uploadType);**

**holdingsUploader = uploadGatewayMap.get(uploadType); holdingsUploader.fetchHoldings (event);**

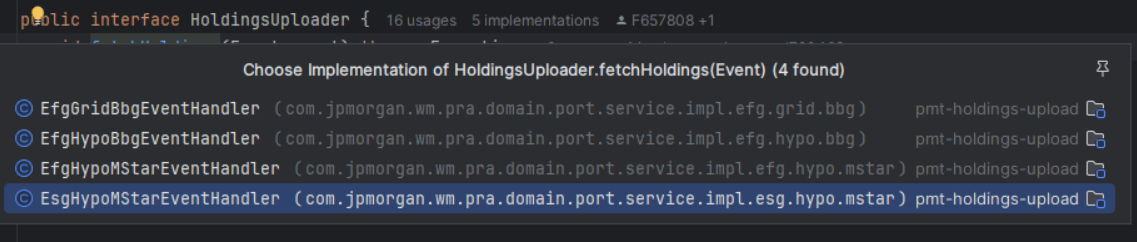
**} catch (Exception e) {**

**log.error("Exception in PMT-HOLDINGS-UPLOAD Lambda");**

**throw new RuntimeException(e);**

**}**

**}**

****

**This details are fetched from Aurora database , Table : ‘GRID\_BBG\_CONFIG’**

**Class GridBbgUploadConfig{**

**gridId**

**gridBaseName (short portfolio name)**

**subcategory ( EQ,FI,AL )**

**accountType**

**cashProxy**

**}**

**Override\_Instruments\_Assets\_Class{**

**Vehicle\_figi\_id**

**Overrride\_asset\_class\_level1**

**isActive;**

**}**

**Map<FigiId,ClassificationLevel> sachClassificationMap**

**Map<FigiId, Overrride\_asset\_class\_level1> overrideAssetClassLevel1Map**

**public class PortfolioHoldings {**

**private String lob;**

**private String accountFigild;**

**private String accountCode;**

**private String accountName;**

**private String accountType;**

**private String branch;**

**private String portfoliold;**

**private String gridId;**

**private String portfolioName;**

**private String inspectionDate;**

**private String asOfDate;**

**private List<Holding> holdings;**

**}**

**public class Holding {**

**private BigDecimal weight;**

**private String closeDate;**

**private long quantity;**

**private BigDecimal marketValue;**

**private BigDecimal price;**

**private Vehicle vehicle;**

**}**

**public class Vehicle extends VehicleId<String> {**

**private long vehicleId;**

**private String figild;**

**private String vehicleName;**

**private String currency;**

**private String cusip:**

**private String isin;**

**private String ticker;**

**private ProductType productType;**

**private AssetClass assetClass;**

**private Classification Level sacClassification;**

**}**

**[Holding (weight=0.22416, closeDate=2024-08-09, quantity=45955, marketValue=8788620.8502, price-191.24, vehicle-Vehicle (vehicleId=11819, figild-BBG001P8F023, vehicleName-FIDELITY 500 INDEX FUND, currency-USD, cusip-315911750, isin-US3159117502, ticker-FXAIX, productType-ProductType (id=301, name=Mutual Fund), assetClass-AssetClass (assetClassId=13, assetClassName=U.S. Large Cap Core Passive Unhedged, classificationLevel-null), sacClassification-ClassificationLevel (level1-Equity, level2-US, level3-Large Cap))),**

**Holding (weight-0.007827, closeDate=2024-08-09, quantity=8828, marketValue-306883.8495, price-34.7604, vehicle-Vehicle (vehicleId=15905, figiId=BBG00J3N6M72, vehicleName=FRANKLIN FTSE CANADA ETF, currency=USD, cusip-35473P827, isin-US35473P8279, ticker-FLCA, productType-ProductType (id=102, name=Exchange-Traded Fund), assetClass-AssetClass (assetClassId=102, assetClassName=Canada Core Passive Unhedged, classificationLevel-null), sacClassification-ClassificationLevel (level1-Equity, level2-International Developed, level3-Canada))),**

**Holding (weight=0.045555, closeDate=2024-08-09, quantity=200007, marketValue=1786062.5323, price-8.93, vehicle-Vehicle (vehicleId=12145, figiId=BBG000BCCCD6, vehicleName=NUVEEN INTRM DUR MUNI BD-I, currency=USD, cusip-67065Q400, isin-US67065Q4001, ticker-NUVBX, productType-Product Type (id-301, name-Mutual Fund), assetClass-AssetClass (assetClassId=357, assetClassName=Core Fixed Income Agnostic muni, classificationLevel=null), sacClassification-ClassificationLevel (level1-Fixed Income, level2-Investment Grade, level3-Municipal))),**

**@able(name = "tbl\_instrument\_proxy\_config")**

**public class Proxy Instrument {**

**@Id**

**@Column(name = "figi\_id")**

**private String figild;**

**@Column(name = "proxy\_figi\_id")**

**private String proxyFigild;**

**@Column(name = "proxy\_ticker")**

**private String proxyTicker;**

**@Column(name = "is\_active")**

**private char isActive;**

**@Column(name = "lob\_type")**

**private String lobType;**

**@Column(name = "suffix")**

**private String suffix;**

**}**