

```

@Service
public class ShipmentFileQuerySideService implements ShipmentFileQuerySide{
    private static final Logger logger = LoggerFactory.getLogger(ShipmentFileQuerySideService.class);
    @Autowired
    private QueryGateway queryGateway;

    @Override
    public boolean FindShipmentFileByFileName(FindShipmentFileByFileNameQuery findShipmentFileByFileNameQuery) {
        logger.info("Dispatch Query {}", findShipmentFileByFileNameQuery);
        CompletableFuture<FindShipmentFileByFileNameQueryResult> findShipmentFileByFileNameQueryResult =
            queryGateway.query(findShipmentFileByFileNameQuery, FindShipmentFileByFileNameQueryResult.class);
        logger.info("Result {}", findShipmentFileByFileNameQueryResult.toString());

        if(findShipmentFileByFileNameQueryResult.join()!=null)
            return true;
        else
            return false;
    }
}

```

```

@Component
public class ShipmentFileQueryHandler {
    private Logger logger = LoggerFactory.getLogger(this.getClass());

    @Autowired
    private ShipmentFileRepository shipmentFileRepository;

    @QueryHandler
    FindShipmentFileByFileNameQueryResult handle(FindShipmentFileByFileNameQuery findShipmentFileByFileNameQuery)
    {
        logger.info("handling {}", findShipmentFileByFileNameQuery);
        Optional<ShipmentFile> shipmentFile = shipmentFileRepository.findByFileName(findShipmentFileByFileNameQuery.getFileName());
        logger.info("return result {}", shipmentFile);
        if(shipmentFile.isPresent())
            return new FindShipmentFileByFileNameQueryResult(shipmentFile.get());
        else
            return null;
    }
}

```