Risks			Baseline Risk Level		
Risk ID	**Cause**	**Description**	**Likelihood**	**Severity**	**Overall Risk Level**
1	Inaccurare estimations	Length of sprint may be underestimated, and work may not be completed within the time frame.	5	5	25
2	Poor quality code	Code may be difficult to read, meaning reviewing it will be difficult and it can be full of bugs. This could result in poor coverage and unsuccessful testing.	6	4	24
3	Poor productivity	Being at home learning a completely new skill may be daunting and could result in poor productivity due to the feeling of being overwhelmed.	3	5	15
4	Resources	Working from home can mean there may be technical failures, such as, no internet connection and failure of hardware, which could prevent the task from being completed.	2	4	8
5	illness	Due to the circumstances and the risk of falling ill with coronavirus and other illnesses, this may prevent work being done due to falling sick and being physically unable to do the work that is required.	1	5	5
6	Expectations	Having high expectations in developing and carrying the project could lead to a feeling overwhelmingness which may result in poor productivity and lack of motivation to commit to the work.	4	5	20
7	Plagarism	Copying work from a fellow peer, which hinders the learning of the one carrying out the project and may result in exclusion from the program.	2	5	10

8	Data Loss	Data loss is always a possibility, may involve losing all work done	1	5	5
9	Database changes	Alterations to Database structure after being populated with records may result in widespread errors.	2	5	10
10	Poor testing coverage	lack or sub standard testing will result in bugs and problems going unoticed	5	1	5

Response		Post-mitigation Risk Level		
Strategy	**Mitigation**	**Likelihood**	**Severity**	**Overall Risk Level**
Reduce	Carry out the immediate tasks at hand first and plan what can and needs to be done during the day, prioritising the most important tasks.	3	4	12
Reduce	Ensure Object Orientated programming principles are followed and format code correctly to ensure it is at a high level of readability.	1	3	3
Reduce	Ask for help when needed from trainers and peers alike and ensure all tasks are attempted using both the resources provided and external sources	2	4	8
Reduce	Notify management immediately if there are technical failures and keep them up to date and ensue that they are resolved as soon as possible.	1	2	2
Reduce	Good project planning (incl. Burndown charts, intermediary deadlines, etc.) will reduce the likelihood of missing the final project deadline.	1	5	5
Reduce	Carry out tasks in small increments and review them. Ensure to not think too far ahead and only focus on the tasks that are being done at present on the day and carry them out to the best of abilities. Use MVP (minimum viable product) criteria to ensure to that the minimum is completed before exceeding expectations.	3	5	15
Reduce	Attempt to understand tasks and review what has been learnt through the training process. Apply the skills to the task and ask for help when needed but do not blindly follow someone.	1	5	5

Reduce	Always push anywork done back to repository frequently and ensure it is backed up online and offline.	1	2	2
Reduce	Plan thoroughly in advance ensuring the databse models follow the ERD and UML that was created initially.	2	1	2
Reduce	set up adequate testing that is in line with the brief	4	1	4

Output	Review
Time was spent planning what will happen and when, given clear time frames and reasonable objectives	This was successful to an extent as most of the tasks that needed to be done was completed within the time frame. However some tasks was not done, due to lack of knowledge rather than time frame.
followed OOP principles and demonstrated some aspects of following SOLID principles aswell	The mimimum MVP was not entirely met as the order CRUD function was not completed so it was difficult to demonstrate OOP comprehensivly. This was due partly to lack of knowledge and understadning as much of the training for java was done very quickly going throug vasts amount of content in a short period
Followed the plan initially created and stayed on task. Asked the trainers for help on aspects of the program	This was successful in terms of attempting everything and trying to understand the program better. However ultimatly could not figure out how to implent the order crud funtion correctly and spent a long time working it out unsuccessfully.
Everything needed was accessible	There was not much issue regarding this aspect as all my resources were availabe. However there was one instance where power was cut due to planning in the building, although this was towards the end of the project and did not have much impact overall.
liklehood of this is difficult to mitigate and and reduce	No illness impacted developer that affected the task at hand.
Some aspects of the crud was completed and some aspects of testing	Overall the developer expectations was not met as the Order CRUD worked but not in the way that it was required. Testing of the item classes were also done that had sufficient coverage but as much time was focused on the Order CRUD not much was done with regards to testing.
Liklehood of this is low due to the ramifications it would have	Asked my peers and trainers alike for some guidance when needed and discussions to generate ideas that would have helped, but did not review anyones code or share my own as it was an independent task therefore mitigating this risk was successful.

Use git bash to push to remote repository	Constantly saved any changes done and whenever a new method or class was implemented this would then be pushed using gitbash to the feature branches. When the feature branch was adequetly completed it would then be merged into the develop. However one issue arose due to misunderstanding beginning where forking was not done properly, however this was then rectified.
Database kept relativly simple and easy to understand	Database was not changed much. Although there some minor changes to ensure the MVP was done so some parts were exluded but they were not significant and changes are shown in the updated ERD
Created testing for the items based on what was done for customers	Only the items testing was done so overall the test coverage is very poor, this is due to the time spent trying to work out how to implent sufficient order CRUD functions