

Quest Pools

You can create a Quest Pool by adding a `*.pool.yml` file into any of the quest folders. The path references like `this` will also be replaced in a Quest Pool like in any other quest. [Quest Pools](#) must specify a trigger in order to be executed. [Quest pools](#) have several options to allow filtering what quests should be given to the player. For that random tables are used.

If a quest in the quest pool is not repeatable it will only be given as long as it is not completed. Repeatable quests are only given after their individual cooldown has passed.

Lets have a look at an example quest pool:

Source Code: `example.pool.yml`

```
1. name: Daily Quests
2. # players will not get new quests from this pool but can still complete existing quests
3. enabled: true
4. # how many quests should maximal be given the player when the trigger is called
5. count: 1
6. # how many active quests can the player maximal have from this quest pool
7. max-active-quests: 1
8. # if set to true will abort all active quests of this pool when the trigger is called and the cooldown is over (and the
   reset time is passed)
9. abort-previous-quests: false
10. # in seconds - how long should be wait until giving new quests
11. cooldown: 3600
12. # time of the day to reset the quest pool and give out new quests - can be left out to only respect the cooldown
13. # e.g.: a reset time of 17:00 and a cooldown of 3600s (1h) will give the player a new quest if it is after 17:00
14. # if the player completes the quest before 0:00 he will get a new quest
15. # if it is after 0:00 he has to wait until 17:00 to get a new quest
16. # so generally a cooldown of 86400 (24h) or more is recommended when using the reset time
17. reset-time: '17:00'
18. # see the action api for a tutorial on how to use trigger and actions
19. # you dont need a action, the quest pool will activate automatically when the trigger is called
20. trigger:
21. 0:
22. type: player.login
23. # delay the trigger by 10s
24. delay: 10
25. # these actions will be executed additionally to the normal quest reward actions
26. # specify requirements to filter for higher rewards on successive daily quests
27. # keep in mind that if you have a successive count requirement with 2 and one with 3, both will be executed
28. # if the player has at least 3 successive quest pool counts
29. actions:
30. 0:
31. type: player.give.money
32. args:
33. amount: 20s
34. requirements:
35. 0:
36. type: questpool.successive.count
37. args:
38. count: 2
39. pool: this.example
40. # now comes the important part - the quests
41. quests:
42. 1:
```

- 43. quest: 'this.path.to.the.first-quest'
- 44. probability: 5
- 45. 2:
- 46. quest: 'this.path.to.the.second-quest'
- 47. probability: 10
- 48. 3:
- 49. quest: 'this.path.to.the.third-quest'
- 50. probability: 1
- 51. 4:
- 52. quest: 'this.path.to.the.fourth-quest'
- 53. # set this to disable the quest IN THE POOL
- 54. enabled: false
- 55. probability: 1

Display All