```
[ceph: root@node /]# ceph osd getcrushmap -o ./map.bin
[ceph: root@node /]# crushtool -d ./map.bin -o ./map.txt
[ceph: root@node /]# cat ./map.txt
...output omitted...
rule replicated_rule {  ❶
        id 0  ❷
        type replicated
        min_size 1  ❸
        max_size 10  ❹
        step take default  ❺
        step chooseleaf firstn 0 type host  ❻
        step emit  ❼
}
...output omitted...
```

❶ The name of the rule. Use this name to select the rule when creating a pool with the `ceph osd pool create` command.

❷ The ID of the rule. Some commands use the rule ID instead of the rule name. For example, the `ceph osd pool set` *`pool-name`* `crush_ruleset` *`ID`* command, which sets the rule for an existing pool, uses the rule ID.

❸ If a pool makes fewer replicas than this number, then CRUSH does not select this rule.

❹ If a pool makes more replicas than this number, then CRUSH does not select this rule.

❺ Takes a bucket name, and begins iterating down the tree. In this example, the iterations start at the bucket called `default`, which is the root of the default CRUSH hierarchy. With a complex hierarchy composed of multiple data centers, you could create a rule for a data center designed to force objects in specific pools to be stored in OSDs in that data center. In that situation, this `step` could start iterating at the data center bucket.

❻ Selects a set of buckets of the given type (`host`) and chooses a leaf (OSD) from the subtree of each bucket in the set. In this example, the rule selects an OSD from each host bucket in the set, ensuring that the OSDs come from different hosts. The number of buckets in the set is usually the same as the number of replicas in the pool (the pool size):

- If the number after `firstn` is 0, choose as many buckets as there are replicas in the pool.
- If the number is greater than zero, and less than the number of replicas in the pool, choose that many buckets. In that case, the rule needs another `step` to draw buckets for the remaining replicas. You can use this mechanism to force the location of a subset of the object replicas.
- If the number is less than zero, subtract its absolute value from the number of replicas and choose that many buckets.

❼ Output the results of the rule.

For example, you could create the following rule to select as many OSDs as needed on separate racks, but only from the DC1 data center:

```
rule myrackruleinDC1 {
        id 2
        type replicated
```