

# API

解锁器API大部分相同就不多描述了，都是用.来call，TTOC.API()不行就试试TTOC:API()

- local TTOC = ... 获得API
- object = TTOC:GetPlayer() 获得角色资讯

```
-- 属性太多，请自行输出察看
local player = TTOC:GetPlayer()
for k, v in pairs() do
    if type(v) ~= "function" then
        print(k, '=', v)
    end
end

--常用function
player:IsCasting() -- 是否战斗中
player:HasAura(idOrName) -- has buff or debuff
player:DistanceTo(x, y, z) --3D距离
player:DistanceTo2D(x, y, z)
player:Jump()
player:Stop()
player:AnimTier() -- The state: 0 = stand, 1 = sit, 3 = sleep
```

- table = TTOC:GetBuffs() 获得角色buffs
- table = TTOC:GetSpells() 获得角色技能

```
-- ex1
GetSpells()[117]:Cast([object])
-- ex2
local spells = GetSpells()
spells[117]:Cast([object])
```

- table = TTOC:GetBag() 获得背包物品资讯

```
local Drink = GetBag().ItemType.Food
Drink[1]:Use()
local mount = GetBag().ItemType.mount
local Armor = GetBag().ItemType[4]
local Weapon = GetBag().ItemType[2]
GetBag().All[113]:Use()
```

- TTOC:OpenAllBag() 打开所有背包
- TTOC:CloseAllBag() 关闭所有背包
- object = TTOC:GetTarget() 获得目标信息

```
-- 属性太多，请自行输出察看
local Unit = TTOC:GetTarget()
for k, v in pairs() do
    if type(v) ~= "function" then
        print(k, '=', v)
    end
end

-- 常用function
Unit:Face() -- 面向
Unit:Interact() -- 与目标互动
Unit:DistanceTo(x, y, z) -- 目标到某处的距离
Unit:HasAura(idOrName) -- 有buff or debuff
Unit:IsQuest() -- 是任务
Unit:GetDistance([OtherUnit]) -- 预设是与玩家距离
Unit:CanLoot() -- 可检取
Unit:CanSkin() -- 可剥皮
```

- bool = TTOC:HasFoodBuff() 是否有Food buff
- bool = TTOC:HasDrinkBuff() 是否有Drinbuff
- TTOC:StartScriptBot() 启动脚本
- TTOC:StopScriptBot() 停止脚本
- TTOC:SetSettings(Settings) 设定

```
TTOC:SetSettings({UseMount = false})
TTOC:SetSettings({
    -- 自动学天赋
    LearnTalent = true,
    -- 使用食物
    UseFood = true,
    -- 使用饮料
    UseDrink = true,
    -- 开启自动休息，不吃药
    AutoRest = false,
    -- 找怪范围
    SearchRadius = 100,
    -- 不攻击等级低于自身等级多少以下的怪物
})
```

```

NotAttackFewLevelsMonster= 5,
--开启使用坐骑
UseMount=true,
--不走水路,假是不走水路,真是走水路
WalkWater =false,
--自动拾取
AutoLoot = true,
--勾选后,所有食物都吃,把包内食物/饮料吃完才会触发城,不用设定FoodName与DrinkName
EatAll = false,
--是否判断回城事件
GoHomeEvent = true,
--恢复生命值范围
FoodPercent = {70,90},
--恢复法力值范围
DrinkPercent = {85,90},
--使用炉石
UseHearthStone = false,
--当背包空格小于等于多少时触发回城
MinFreeBagSlotsToGoToTown = 2,
--当装备耐久度小于等于%多少时触发回城
MinDurabilityPercent = 20,
--当弹药数量少于多少时触发回城
AmmoAmountToGoToTown = 0,
--贩卖颜色等级0~8
SellLevel = {[0] = true,[1] = true,[2] = true},
--不贩卖列表 {id,id,id
DoNotSellList = {'Rune of','传送','Hearthstone','炉石','Primal','源生','Mote
of','微粒','Air','空','基尔加丹印记','Mark of Kil'jaeden'},
--强制贩卖列表 {id,id,id
ForceSellList = {'亚麻布','毛料','丝绸','魔纹布','符文布','硬肉干','清凉的泉
水','血岩碎片','森林蘑','Linen Cloth','Wool Cloth','Silk Cloth','Mageweave
Cloth','Runecloth','Tough Jerky','Refreshing Spring Water','Blood
Shard','ForestMushroom Cap'},},
--强制销毁
ForceDeleteList = {'秘教的命令','Cabal Orders' 'OOX'},
--是否反击
Counterattack = true,
--自动换装
AutoEquip = true,
--购买最好
Buybest = false,
--食物数量
FoodAmount = 0,
--饮料数量
DrinkAmount = 0,
})

```

- value = TTOC:Profile(name) 获得设定内容

```

TTOC:Profile('FoodAmount')
TTOC:Profile('AutoEquip')
TTOC:Profile('UseMount')
TTOC:Profile('DoNotSellList')

```

- `bool = TTOC:ReleaseCorpse()` 释放尸体
- `bool = TTOC:RecoverCorpse()` 确认复活
- `count, table = TTOC:GetQuestInfo()` 获得任务数量与相关信息
- `bool = TTOC:IsHasQuest(QuestID)` 是否接过任务
- `bool = TTOC:IsCompletedQuest(QuestID)` 是否完成过任务
- `object = TTOC:FindUnit(CallFunc[,sortFunc])` 遍历单位取一个
- `table = TTOC:FindUnits(CallFunc[,sortFunc])` 遍历单位取多个

```

local obj = TTOC:FindUnits(
    function(o)
        if (o.ObjectID == 6491) then return true end
    end)

```

- `object = TTOC:FindGameObject(CallFunc)` 遍历物体取一个
- `table = TTOC:FindGameObjects(CallFunc)` 遍历物体取多个
- `object = TTOC:FindNPC(CallFunc)` 遍历NPC取一个
- `TTOC:FindPlayer(CallFunc)` 遍历玩家取一个
- `TTOC:FindPlayers(CallFunc)` 遍历玩家取多个
- `TTOC:FindAll(CallFunc)` 遍历所有取多个
- `object = TTOC:GetNPCInfo(NPCID)` 获得NPC信息，NPC要在附近，多个符合就挑最近
- `bool = TTOC:UnitInWater(Object)` 目标是否在水中
- `bool = TTOC:UnitInvisible(Object)` 是否可见
- `bool = TTOC:PathIsBlocked(ax,ay,az,bx,by,bz)` 两点是否是可通过
- `bool = TTOC:PathInWater(x,y,z)` 坐标是否在水中
- `count = TTOC:GetFoodCount()` 食物数量

- `count = TTOC:GetDrinkCount()` 饮料数量
- `count = TTOC:GetProjectileCount(type)` 弹药数量

```
--[[
type
0 杖 ( 过时 )
1 个螺栓 ( 过时 )
2 箭
3 子弹
4 投掷 ( 作废 )
]]
```

- `value = TTOC:GetDurabilityPercent()` 装备耐久
- `TTOC:BuyItem(ItemId[,count])` 购买商品，`count`不填预设为商品数量
- `TTOC:DeleteItem(ItemId)` 销毁物品
- `TTOC:UseItem(ItemId)` 使用物品
- `bool = TTOC:IsInDungeon()` 是否在地牢
- `bool = TTOC:IsDead()` 是否死亡
- `TTOC:NPCRetrieve(Bots)` 医者虚弱复活
- `TTOC:AllLoot(Bots)` `all loot`

```
local Bots = TTOCInitScript("冲冲冲")
-- ...
TTOC:AllLoot(Bots)
```

- `TTOC:FacingTagert()` 面向目标
- `bool = TTOC:IsCombating()` 是否战斗中
- `TTOC:CombatingPrint(bool)` 战斗通知开关
- `TTOC:SetNoCombating(Bots, Seconds)` 设定非战斗中

```
local Bots = TTOCInitScript("冲冲冲")
-- ...
RunMacroText("/stopcasting")
SetNoCombating(Bots, 6)
```

- `bool = TTOC:IsKnowSkill(SkillID)` 会这个技能
- `TTOC:SetNoClip(NoClip)` 启用穿墙，会自动启动飞天避免掉落

```
--[[
NoClip
    [0] = '关闭 ',
    [1] = '建筑 ',
    [2] = '静态对象',
    [4] = '动态对象-例如门',
    [8] = '地形',
    [15] = '所有',
]]
SetNoClip(15)
SetNoClip(0)
```

- `TTOC:SetFloating(bool,Pitch)` 启用升天

```
--[[
Pitch
    [3] = '上升',
    [2] = '平行',
    [1] = '下降'
]]

TTOC:SetFloating(true, 2)
TTOC:SetFloating(false)
```

- `TTOC:MoveToZ(Z)` 上升
- `TTOC:FlyingTo(X, Y, Z,LevelFly)` 飞到(x,y,z)，要开飞天
- `TTOC:LearnPetSpell(id)` 学习宠物技能

```
TTOC:LearnPetSpell(2649)
TTOC:LearnPetSpell(14916)
```

- `bool = TTOC:BehindIsWall()` 背后是否是墙

- `bool = TTOC:BehindHasHole()` 背后是否有坑
- `bool = TTOC:HasMount()` 是否拥有坐骑
- `TTOC:UseMount()` 使用坐骑
- `bool = TTOC:HasFlyMount()` 是否拥有飞天骑宠
- `TTOC:UseFlyMount()` 召唤骑宠飞天
- `bool = TTOC:InAzeroth()` 是否在艾泽拉斯
- `bool = TTOC:CanGoTo(x, y, z)` 能否移动到该坐标