

rust 编程要点

单元测试

演示文件测试

#重载

#继承

重载函数 和 继承

```
fn main() {
    let x=Node::new(1);
    let y=Node::new((1,2));
    print(&x);
    print(&y);
}
#[derive(Debug)]
struct Node{
    x:i32,
    y:i32,
}
trait new<T> {
    fn new(value:T) ->Self;
}
impl new<i32> for Node {
    fn new(value:i32) ->Self {
        Node { x: value, y: value }
    }
}
impl new<(i32,i32)> for Node {
    fn new(x:(i32,i32))->Self{
        Node{x:x.0,y:x.1}
    }
}
trait X_Y {
    fn print(&self);
}
impl X_Y for Node {
    fn print(&self){
        println!("{}",self.x,self.y);
    }
}
fn print<T:X_Y>(x:&T){
    x.print();
}
//或
```

```
fn print(x:&dyn X_Y){  
    x.print();  
}
```