

Generics

why

To reuse code in a type safe way

Generics

why

To reuse code in a type safe way

Generics

```
fun Account.withdraw(amount: BigDecimal): Pair<Account, BigDecimal> = when (this) {  
    is CurrentAccount -> copy(balance = balance - amount) to amount  
    is SavingsAccount -> copy(balance = balance - amount) to amount  
    is CreditAccount ->  
        if (balance - amount > creditLimit) copy(balance = balance - amount) to amount  
        else this to BigDecimal.ZERO  
    is TechnicalAccount -> this to BigDecimal.ZERO  
}
```

```
val account = SavingsAccount("GB00...", 10_000.toBigDecimal(), "0.024".toBigDecimal())  
println(account.interest)
```

```
val (updatedAccount, _) = account.withdraw(10.toBigDecimal())  
println(updatedAccount.interest)
```