```kotlin
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
}
```

Generics

```
fun Account.withdraw(amount: BigDecimal): Pair<Account, BigDecimal>
```

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

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

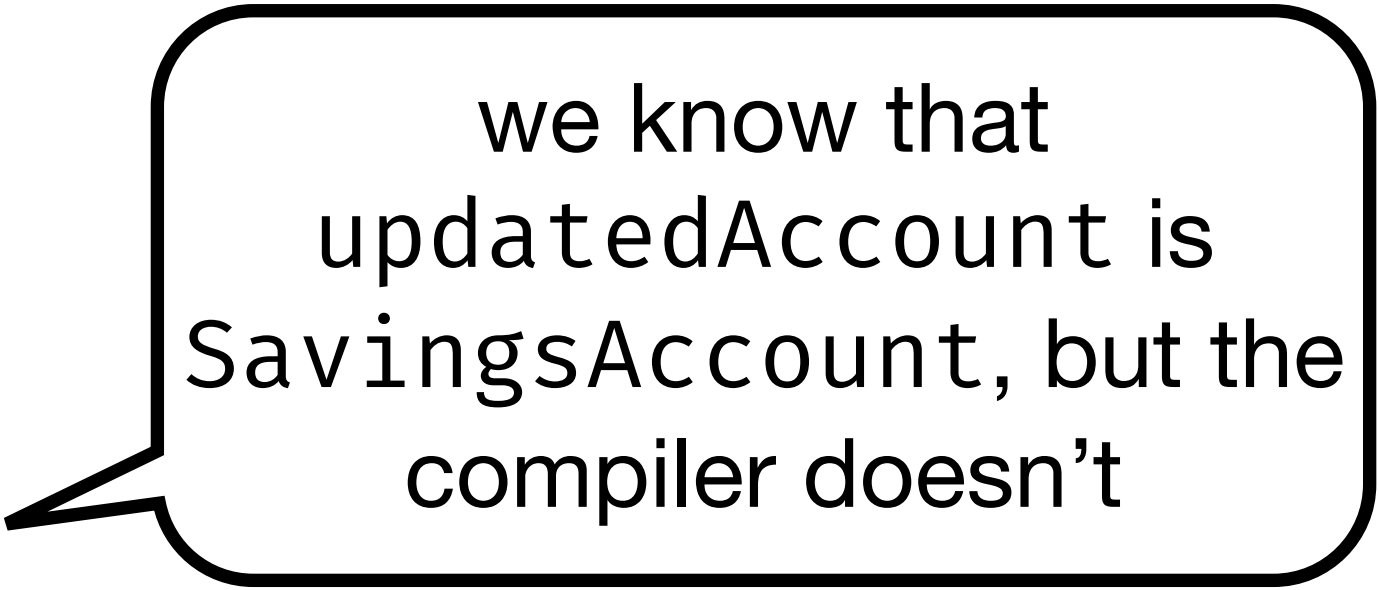
compilation error: Unresolved reference: interest

```
val account = SavingsAccount
```

account.**interest**

```
val  updatedAccount      = account.withdraw
```

```
updatedAccount.interest
```


we know that
updatedAccount is
SavingsAccount, but the
compiler doesn't

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)
```

compilation error: Unresolved reference: interest

we know that updatedAccount is SavingsAccount, but the compiler doesn't

Generics

to save the day

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

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