Code: Important part for us but also of this script is validation. The borrower gets his personal private key / code that he can use to redeem funds. He will be able to unlock the funds when valid.

The *Gift.hs* script can be integrated in the marketplace. The Haskell function mkValidator (1) that represents the validator can be modified. The function consists of three arguments: Datum, Redeemer and ScriptContext. These share the same data types called data.

(1) mkValidator :: Data → Data → Data → ()

(2) mkValidator :: () → Integer → ScriptContext → Bool

The modfied mkValidator (2) is an example. The lender can give the borrower the code to unlock the funds in the Marketplace. In this example the borrowers code is represented by an Integer.

Custom data types can be used too and we did not talk about the ScriptContext. There are lots of possibilities with the validation script to transfer or donate funds without the use of a Middleman and this keeps human error to a minimum.