```
const validator = (function () {
    let isValid = false;

    function invalid() {
        return isValid = false
    }
    return {
    isNumeric: function (text) {return Number.isNaN(Number(text)) ?
invalid() : true;},
    isInteger: function (text) {return
Number.isInteger(Number(text)) ? true : invalid();},
    isNegativeInteger: function (text) {let value =
Number(text);return value < 0 && Number.isInteger(value) ? true :
invalid();},
    isPositiveInteger: function (text) {let value =
Number(text);return value > 0 && Number.isInteger(value) ? true :
invalid();},
    isNonNegativeInteger: function (text) {let value =
Number(text);return value >= 0 && Number.isInteger(value) ? true :
invalid();},
    isInRange: function (text, m, n) {let value = Number(text);return
value >= m && value <= n ? true : invalid();},
    isValidEmail: function (text) {return  /^(([^<>()\[\]\\.,;:\s@"]+
(\.[^<>()\[\]\\.,;:\s@"]+)*)|(".+"))@((([0-9]{1,3}\.[0-9]{1,3}\.[0-9]
{1,3}\.[0-9]{1,3}])|(([a-zA-Z\-0-9]+\.)+[a-zA-Z]{2,}))$/},
    isNonEmpty: function (text) {text.length >0 ? true : invalid();},
    matchesRegex: function (text, regex) {return typeof text ===
"string" && text.match(regex) ? true : invalid();},
    lengthIsInRange: function (text, m, n) {let value =
Number(text);return value >= m && Number.isInteger(value) && value <=
n ? true : invalid();},
    isValid: function () {return isValid},
    reset: function () {isValid = true;},
    };
   }());

   validator.reset(); // Write some code to test your solution
   if (validator.isValid()) {console.log('All is well');} else
{console.log('Something failed validation'); }
   validator.isNumeric(5);
   if (validator.isValid()) {console.log('All is well');} else
{console.log('Something failed validation'); }
   validator.isNumeric('5');
   if (validator.isValid()) {console.log('All is well');} else
{console.log('Something failed validation'); }
   validator.isNumeric('five');
   if (validator.isValid()) {console.log('All is well');} else
{console.log('Something failed validation'); }
```

```
   validator.reset(); // Write some code to test your solution
   validator.isNegativeInteger('five');
   if (validator.isValid()) {console.log('All is well');} else
{console.log('Something failed validation'); }
   validator.reset(); // Write some code to test your solution
   validator.isNegativeInteger('5');
   if (validator.isValid()) {console.log('All is well');} else
{console.log('Something failed validation'); }
   validator.isNegativeInteger(-3);
   if (validator.isValid()) {console.log('All is well');} else
{console.log('Something failed validation'); }
   validator.reset(); // Write some code to test your solution
   validator.isNonNegativeInteger(0);
   if (validator.isValid()) {console.log('All is well');} else
{console.log('Something failed validation'); }
   validator.reset(); // Write some code to test your solution
   validator.isNonNegativeInteger(1);
   if (validator.isValid()) {console.log('All is well');} else
{console.log('Something failed validation'); }
```