Sử dụng prompt

npm install prompt

Usage :

const prompt = require("prompt");

prompt.start();

Without Validation

/\* No Schema \*/

prompt.get(["username", "email"], function (err, result) {

  console.log("username: " + result.username);

  console.log("email: " + result.email);

});

With Validation

var schema = {

  properties: {

    name: {

      pattern: /^[a-zA-Z\s\-]+$/,

      message: "Name must be only letters, spaces, or dashes",

      required: true,

    }

  }

};

/\* With Schema \*/

prompt.get(schema, function (err, result) {

  console.log("  name: " + result.name);

});

Validation

{

    description: 'Enter your password',

    type: 'string',

    pattern: /^\w+$/,

    message: 'Password must be letters',

    hidden: true,

    replace: '\*',

    default: 'lamepassword',

    required: true,

    maxLength: 8,

    minLength: 8,

    minimum: 10,

    maximum: 10,

    divisibleBy: 5,

    minItems: 2,

    maxItems: 2,

    format: '...',

    conform: function (v) {

        if (v%3==1) return true;

        return false;

    },

    before: function(value) { return 'v' + value; }

}

Hidden : thường sd cho password (không hiển thị ký tự khi đang nhập)

Replace : nếu “hidden” = true 🡪 ~ khoảng trắng sẽ được thay = replace’s value

Before : là middleware thay đổi giá trị sau khi input thành công

Type : string, boolean, number, integer, array : khi specify type thì nó tự type cast

Boolean gồm : 'true', 't', 'false', 'f'

Conform : validation theo ý thích

Format : là những type có sẵn

{ format: 'url' }

{ format: 'email' }

{ format: 'ip-address' }

{ format: 'ipv6' }

{ format: 'date-time' }

{ format: 'date' }

{ format: 'time' }

{ format: 'color' }

{ format: 'host-name' }

{ format: 'utc-millisec' }

{ format: 'regex' }

Code template

prompt.get([{

    name: 'email',

    minLength: 8,

    message: 'Must be a valid email with minimum length is 8 chars',

    format: 'email',

    required: true

}], function (err, result) {

    console.log('username: ' + result.username);

});

prompt.get(

  [

    {

      name: "Names(Ctrl C to finish)",

      type: "array",

      required: true,

    },

  ],

  function (err, result) {

    console.log("names: " + result.names);

  }

);

prompt.get(

  [

    {

      name: "password",

      type: "string",

      message: "Password can't leave blank",

      hidden: true,

      replace: "\*",

      required: true,

    },

  ],

  function (err, result) {

    console.log("password: " + result.password);

  }

);

# Sync Prompt

/\* Basic \*/

const prompt = require("prompt-sync")();

var name = prompt("What's your name : ");

console.log(name);

/\* Advanced \*/

require('prompt-sync')({...config}})

var prompt = require('prompt-sync')({

    history: require('prompt-sync-history')(),

    autocomplete: complete(['hello1234', 'he', 'hello', 'hello12', 'hello123456']),

    sigint: false

});

Config props :

sigint: Default is false.

 - Nếu sigint = false, Khi ctrl c thoát khỏi prompt => returns null.

 - Nếu sigint = true => sẽ thoát chương trình.

eot: Default is false. Ctrl D sẽ exit luôn

autocomplete: khi mà nhấn tab thì sẽ autocomplete

history: lưu vào file

Write to File

/\*

    History [npm install --save prompt-sync-history]

\*/

var prompt = require("prompt-sync")({

  history: require("prompt-sync-history")(),

  // OR : history: require("prompt-sync-history")("filename.txt"),

});

var input = prompt();

prompt.history.save(); //save history back to file

Code template

/\* Usage \*/

prompt(ask, value, opts)

* ask: là label
* value: là default value
* echo: Default is '\*'. khi enter pass thì value sẽ được replace = echo's value
  + hoặc sử dụng prompt.hide
* autocomplete: Override [autocomplete] của instance scope. chỉ định cho prompt cụ thể
* value:
  + default value for the prompt.
  + If opts is in the third position, this property will not overwrite the value parameter.
* ask:
  + Sames as the value parameter.
    - The prompt label. If opts is not in the first position, the ask parameter will not be overridden by this property.

var pwb = prompt('enter hidden password (or don\'t): ', {echo: '', value: '\*pwb default\*'})

var pwc = prompt.hide('enter another hidden password: ')

var autocompleteTest = prompt('custom autocomplete: ', {

    autocomplete: complete(['bye1234', 'by', 'bye12', 'bye123456'])

});

Code template

var prompt = require("prompt-sync")();

var pw = prompt("nói gì đi", {

  autocomplete: complete(["dad", "daddy"]),

});

console.log(pw);

function complete(commands) {

  return function (str) {

    var i;

    var ret = [];

    for (i = 0; i < commands.length; i++) {

      if (commands[i].indexOf(str) == 0) ret.push(commands[i]);

    }

    return ret;

  };

}