

# **Ruby 講義**

## **第3回 Ruby入門**

**Kuniaki IGARASHI/igaiga**

**2012.4.26 at 一橋大学**

**社会科学における情報技術とコンテンツ作成III  
(ニフティ株式会社寄附講義)**

お知らせ

来週の講義は  
5/1 (火) です。

Teaching Assistant 濱崎 健吾  
クックパッド株式会社 エンジニア





教科書 p.4 ~ p.21

# やること

前回のおさらい

**irb でRubyプログラムを実行する  
オブジェクト、メソッド、引数  
計算**

**.rb ファイル でRubyプログラムを実行する  
日本語、文字コード、マジックコメント**

# この講義でRubyを動かす方法

1. VM 上の Linux
2. (Windows環境の人) RailsInstaller
- (3. ブラウザから TryRuby)
- (4. mac でがんばってインストール)

# VM環境構築

1. USBメモリから .vhd ファイルコピー
2. 下記のページを参考に起動

<http://www.ubuntulinux.jp/download/ja-remix-vhd>

次ページ、次々ページにポイントを貼ってあります。

id: dev password: devdev

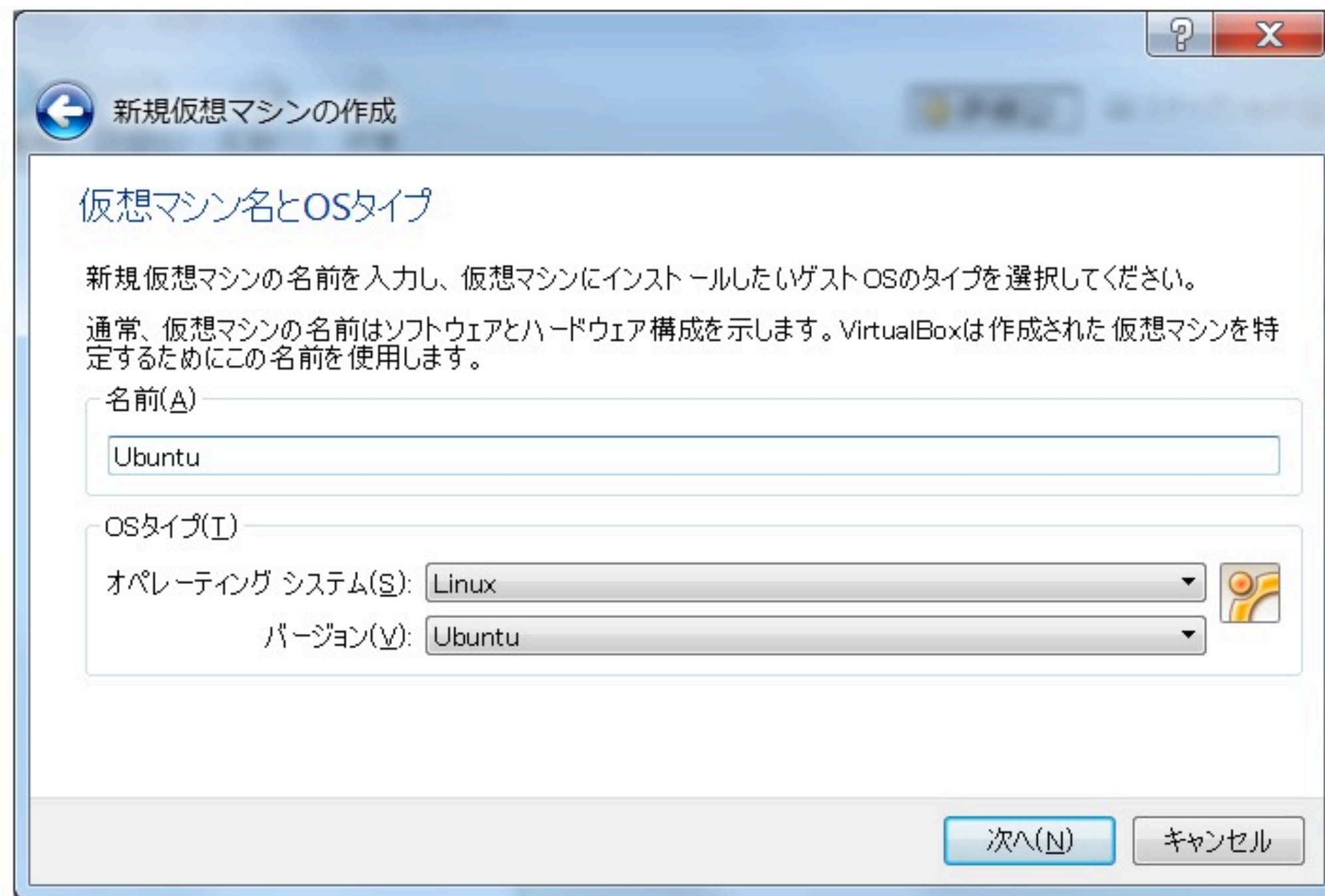
3. 「端末」を起動してrubyを動かす

```
$ irb
```

```
irb(main):001:0> p 'Hello world'
```

## 仮想マシンの登録と起動

VirtualBoxを起動し、[新規]をクリックします。仮想マシン作成ウィザードが開くので、画面の指示に従って入力していきます。OSタイプは「Ubuntu」を選んで下さい。



新規仮想マシンの作成

### 仮想マシン名とOSタイプ

新規仮想マシンの名前を入力し、仮想マシンにインストールしたいゲストOSのタイプを選択してください。

通常、仮想マシンの名前はソフトウェアとハードウェア構成を示します。VirtualBoxは作成された仮想マシンを特定するためにこの名前を使用します。

名前(A)

Ubuntu

OSタイプ(I)

オペレーティング システム(S): Linux

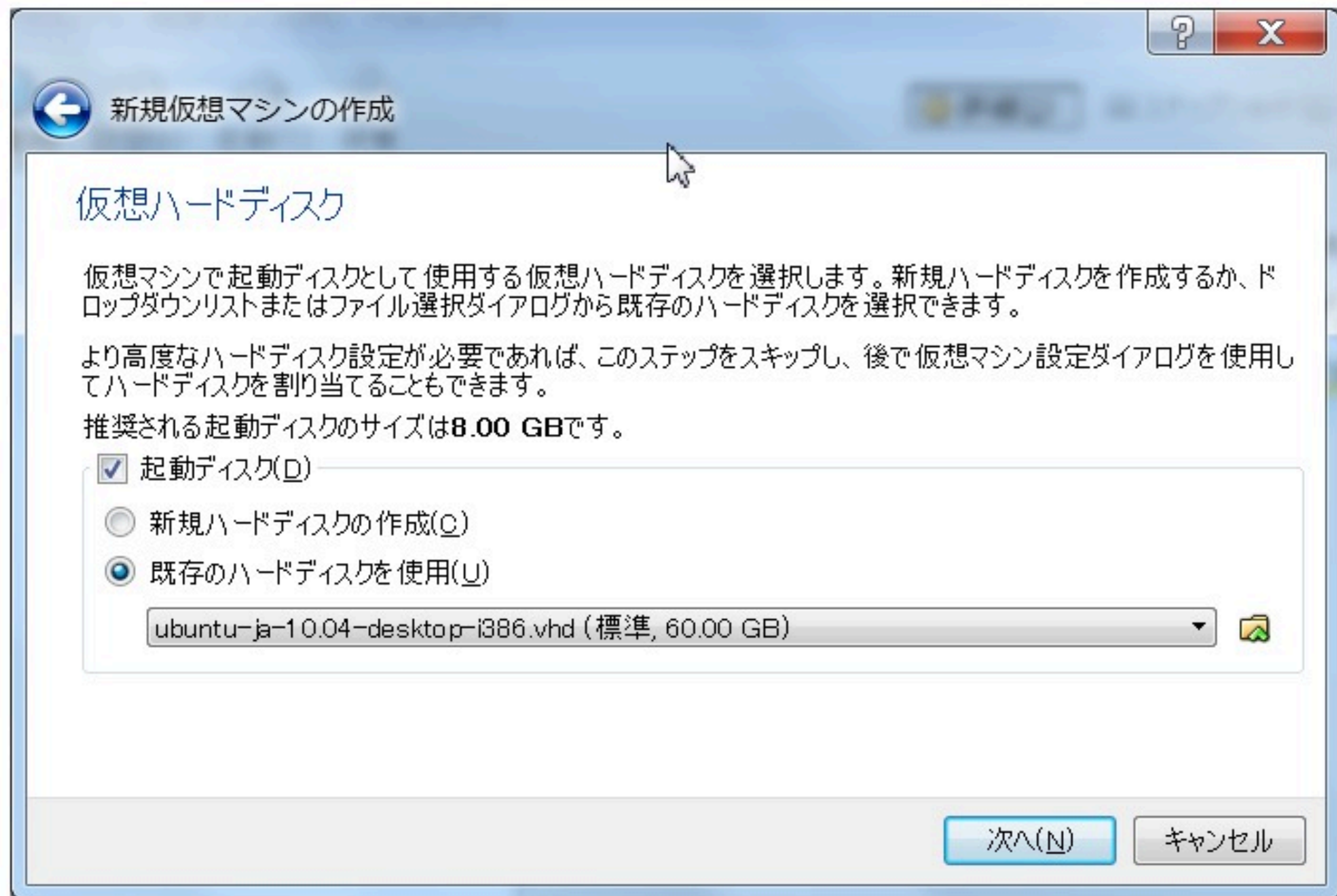
バージョン(V): Ubuntu

次へ(N) キャンセル

<http://www.ubuntulinux.jp/download/ja-remix-vhd>



[仮想ハードディスク]ダイアログでは、展開した仮想マシンイメージを選びます。



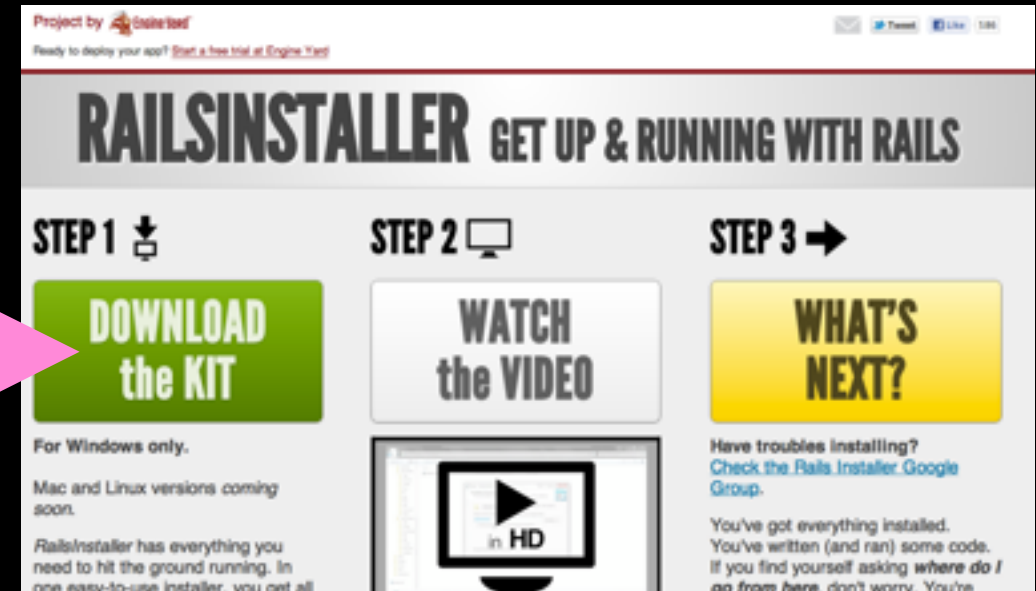
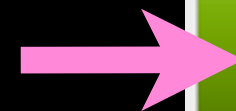
作成後、[起動]をクリックすると仮想マシンが起動します

<http://www.ubuntulinux.jp/download/ja-remix-vhd>



# RailsInstaller (Windows用)

**1. Railsインストーラーを  
Download, Install  
<http://railsinstaller.org/>**



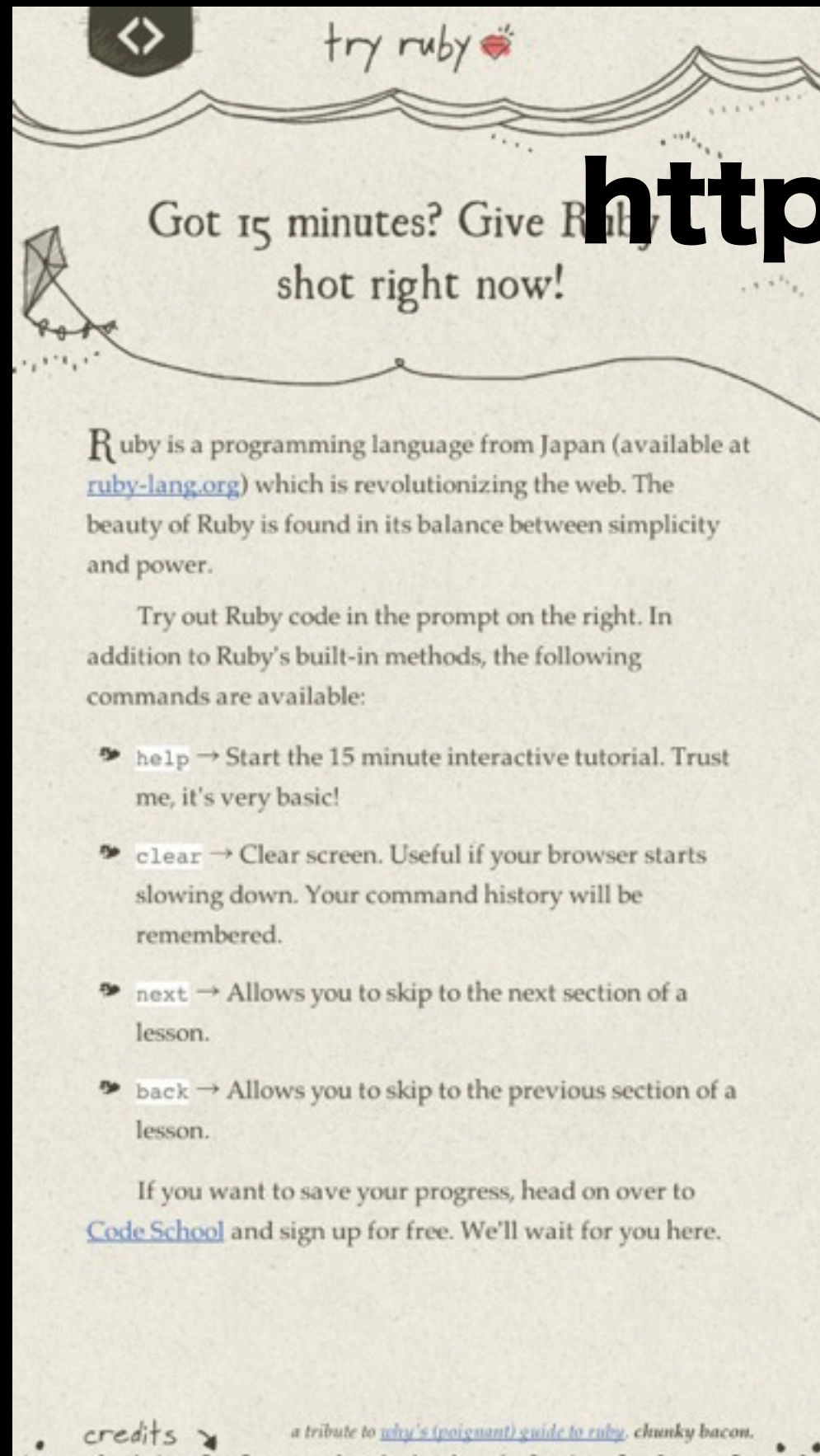
**2. 名前とemailを入力（公開されます。）  
（将来つかう）githubの設定で使われます。**

**3. 起動 Windowsメニューから以下を実行**

**「プログラムとファイルの検索」から "Command Prompt with Ruby  
and Rails" or**

**スタート→全てのプログラム→RailsInstaller→"Command Prompt with  
Ruby and Rails"**

# ブラウザからTryRuby

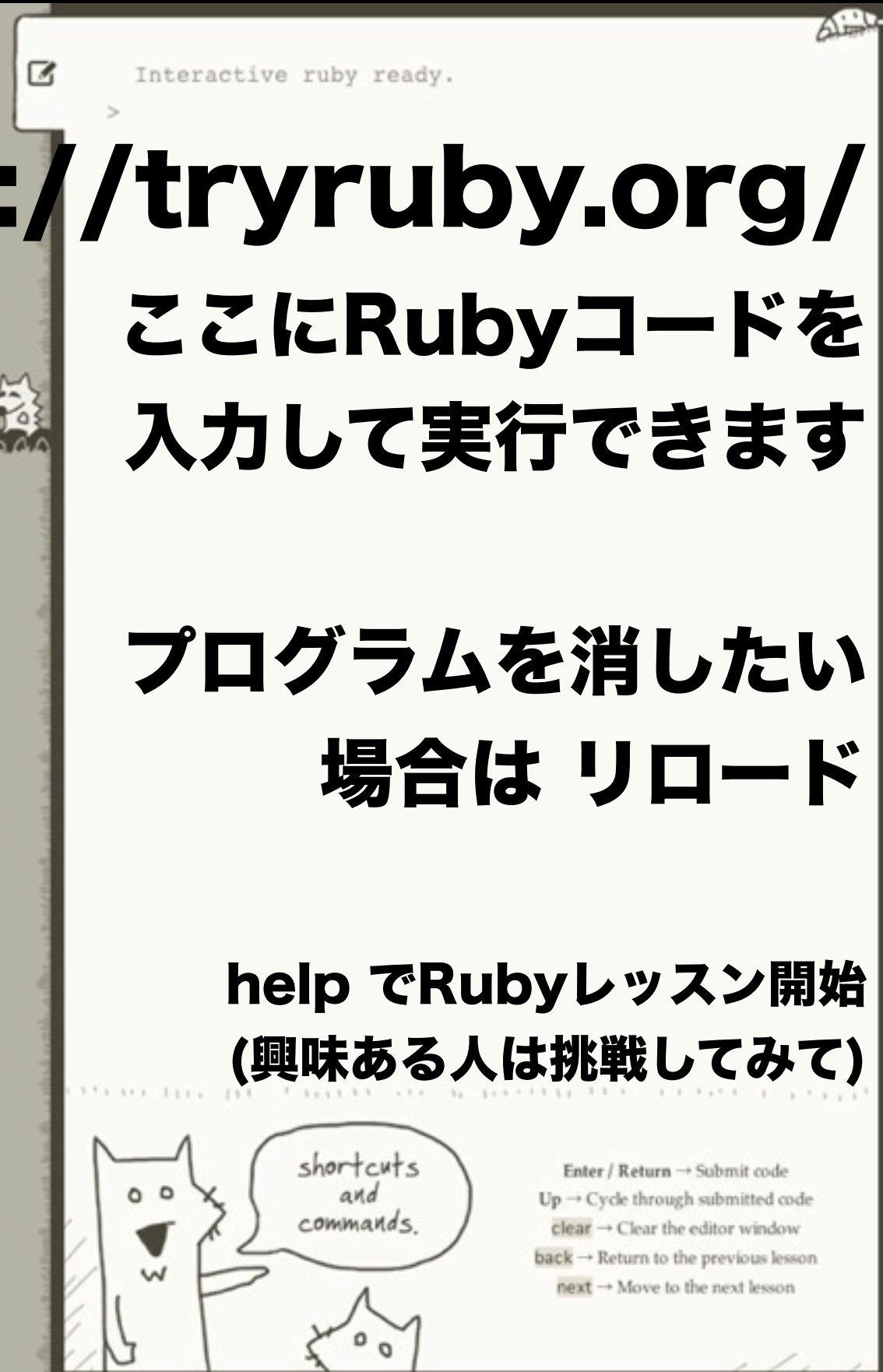


**<http://tryruby.org/>**

**ここにRubyコードを  
入力して実行できます**

**プログラムを消したい  
場合は リロード**

**help でRubyレッスン開始  
(興味ある人は挑戦してみて)**



# Rubyコードを実行する方法

**a. irb から1行ずつ実行**

※**irb** = **Interactive RuBy** の略

**b. ファイル(.rb) にコードを書いて実行**

**言葉の解説：コード = プログラム です。**

**コードって言うとプロっぽくてかっこいいです。**

**まずは**

**irb**

**を使ってみましょう**

# irb 起動方法

## VM Ubuntu

“端末”を起動, irb と入力(次ページ参照)

## RailsInstaller

"Command Prompt with Ruby and Rails" を起動,  
irbと入力

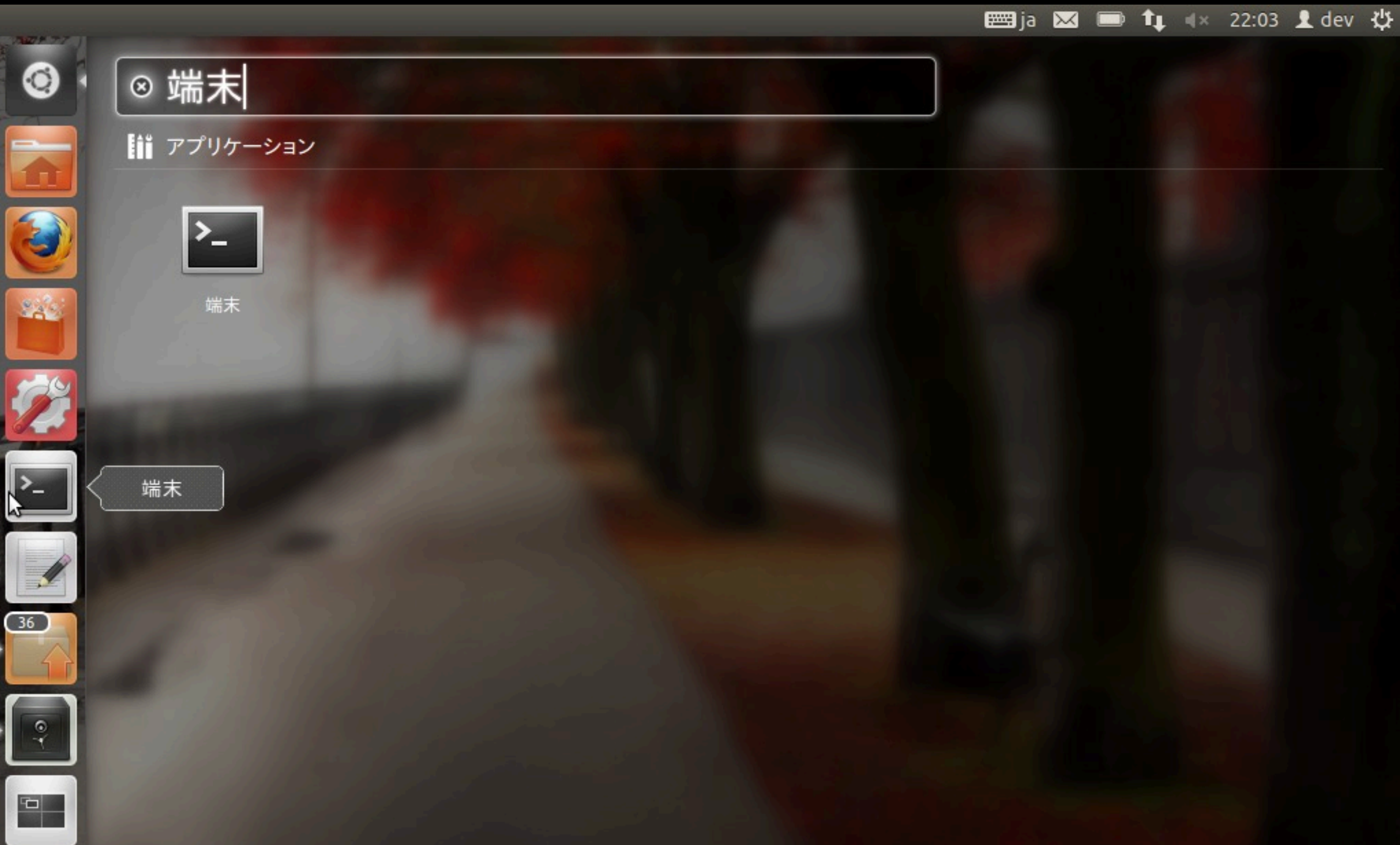
## TryRuby

特に何もしなくて大丈夫です。

そのまま irb みたいに動きます。



# VM Ubuntu





以下のRubyコードを実行  
画面に Hello, Ruby と表示するコード

```
print("Hello, Ruby.\n")
```

※Windowsでは \ は ¥ と打ちます。

なぜかと言うと・・・話すと長くなりますよ？

```
$ irb
```

```
irb(main):001:0> print("Hello, Ruby.\n")
```

```
Hello, Ruby.
```

```
=> nil
```

# よく使う 用語の説明

```
print("Hello, Ruby.\n")
```

オブジェクト

Stringオブジェクト

```
print("Hello, Ruby.\n")
```

**メソッド**

**引数**

**メソッド：手続き、命令**

**引数：メソッドに渡すデータ**

**printメソッドは画面に引数のデータを表示する命令**

**printメソッドに**

**"Hello, Ruby.\n" オブジェクトを**

**引数として渡しています。**

**ちなみに、最後の \n は改行のことです。**

# 画面に表示するメソッド

よく使うので似た機能のメソッドが3つあります。

**print** : 表示(改行しない)

**puts** : 表示(改行する)

**p** : 調査(デバッグ)用

※用語解説 : デバッグ

バグ (不具合) を解消すること

次は  
計算を  
してみましよう



# 以下の計算を試みましょう

**puts 1 + 2**

**puts 2 - 3**

**puts 5 \* 10**

**puts 100 / 4**

**puts 2\*\*32**

**puts 10/3**

**puts 10/0**

**+: 足し算**

**-: 引き算**

**\*: 掛け算**

**/: 割り算**

**\*\* : 累乗**

logとかsinとかもあります。  
知りたい方はこちら。

# 言葉の説明

puts 1 + 2  
puts 2 - 3  
puts 5 \* 10  
puts 100 / 4  
puts 2\*\*32  
puts 10 / 3  
puts 10 / 0

メソッド

puts

演算子

+ とか - とか

オブジェクト

1とか2とか

整数オブジェクト

(Fixnumオブジェクト)

# 計算するときの注意

puts 1 + 2

puts 2 - 3

puts 5 \* 10

puts 100 / 4

puts 2\*\*3

puts 10 / 3 => 3

puts 10 / 0

ところで、これ、  
結果が変だと  
思いませんか？

# 計算するときの注意

# 整数オブジェクト同士の計算は整数オブジェクトになる

> puts 10/3

**$\Rightarrow 3 \leftarrow 3.33\dots$  ではない**

**小数で求めたい場合は両方もしくは片方を  
小数(Float)オブジェクト(例：10.0)にする**

```
> puts 10.0/3.0
```

**$\Rightarrow 3.3333333333333333$**

> puts 10/3.0

**$\Rightarrow 3.3333333333333333$**

# 数値オブジェクト

**Rubyで数値を扱う場合**

**2：整数オブジェクト**

**(Fixnumオブジェクト)**

**3.14：小数（浮動小数点数）オブジェクト**

**(Floatオブジェクト)**

ほかにもあります。そのうちでできます。

# 計算するときの注意

```
puts 1 + 2  
puts 2 - 3  
puts 5 * 10  
puts 100 / 4  
puts 2**3  
puts 10 / 3 => 3  
puts 10 / 0
```

ところで、これ、  
何言ってるんです  
か？

```
ZeroDivisionError:  
divided by 0  
from (irb):2:in `/'  
from (irb):2
```



# エラーメッセージ

正しくないプログラムを実行したり、  
Rubyが処理に困ると、  
エラーメッセージが表示されます。

```
irb(main):001:0> 10/0  
ZeroDivisionError:  
  divided by 0  
    from (irb):2:in `/'  
    from (irb):2
```

日本語訳

ゼロ除算エラーだよ。  
0で割ってるよ。  
/のところだよ。

0 で割り算するとエラーになるので  
注意してください。

**次の話題は  
少しマニアックですが、  
心のどこかに留めて  
おいてください**

# Floatオブジェクトは不正確

0.1を3回足したら0.3になりますよねもちろん

```
irb> 0.1+0.1+0.1 == 0.3
```

=> **false**

**==** は 等しいかどうか調べる演算子

0.1を3回足したら0.3に**ならなかった**

Rubyの中の0.1は正確に数学上の0.1と一致しない

※**整数(Fixnum)**オブジェクトは正確です。

**小数を正確に計算したい場合は、  
整数にして計算するか、  
Rational(有理数)オブジェクトを使う**

**Rational(1/10) +  
Rational(1/10) +  
Rational(1/10) == Rational(3/10)  
=> true**

**1/10を3回足したら3/10になった  
Rational(分母/分子)**

**では、そろそろ  
irb の世界から抜けて  
みましょう。**

# irb の世界と shell の世界

irb の外は shell という世界

```
$ irb
```

```
1.9.3p125 :001 > puts 1+2+3
```

```
6
```

```
=> nil
```

```
1.9.3p125 :002 > exit
```

```
$ pwd
```

```
/Users/igarashi
```

```
$
```



# irb の世界と shell の世界

irb の外は shell という世界

```
$ irb
```

shell

```
1.9.3p125 :001 > puts 1+2+3
```

```
6
```

Ruby語が通じる世界

```
=> nil
```

```
1.9.3p125 :002 > exit
```

irb

```
$ pwd
```

```
/Users/igarashi
```

shell語が通じる世界

OSと話ができる

```
$
```

shell

# 世界を移動するコマンド

```
$ irb
```

```
shell
```

```
1.9.3p125 :001 > puts 1+2+3  
6
```

```
=> nil
```

```
1.9.3p125 :002 > exit
```

```
irb
```

```
$ pwd
```

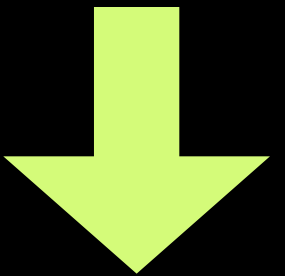
```
/Users/igarashi
```

```
$
```

```
shell
```

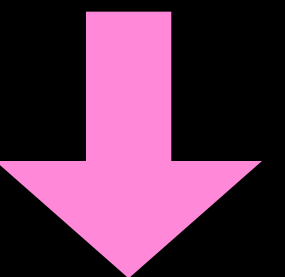
irb

コマンド



exit

コマンド



**次はファイルに  
Rubyコードを  
書いてみましょう。**

Rubyコードをファイルに記述して実行

helloruby.rbを作って保存

```
print("Hello, Ruby.\n")
```

helloruby.rbを実行

```
$ ruby helloruby.rb  
Hello, Ruby.
```

shellでRubyコードが書かれたファイルを実行するには  
ruby ファイル名

# エディタ - コードを打つ道具

推奨エディタ(お気に入りがある人はそれでOKです)

Windows : intype

<http://inotai.com/intype/>

VM Ubuntu : gedit テキストエディター  
(インストール済)

Mac : テキストエディット (インストール済)

CotEditor <http://sourceforge.jp/projects/coteditor/>

# Windows intype

<http://inotai.com/intype/>



The text editor we've all been waiting for.

Blog

Download

Follow us



Highly addictive  
text editing!

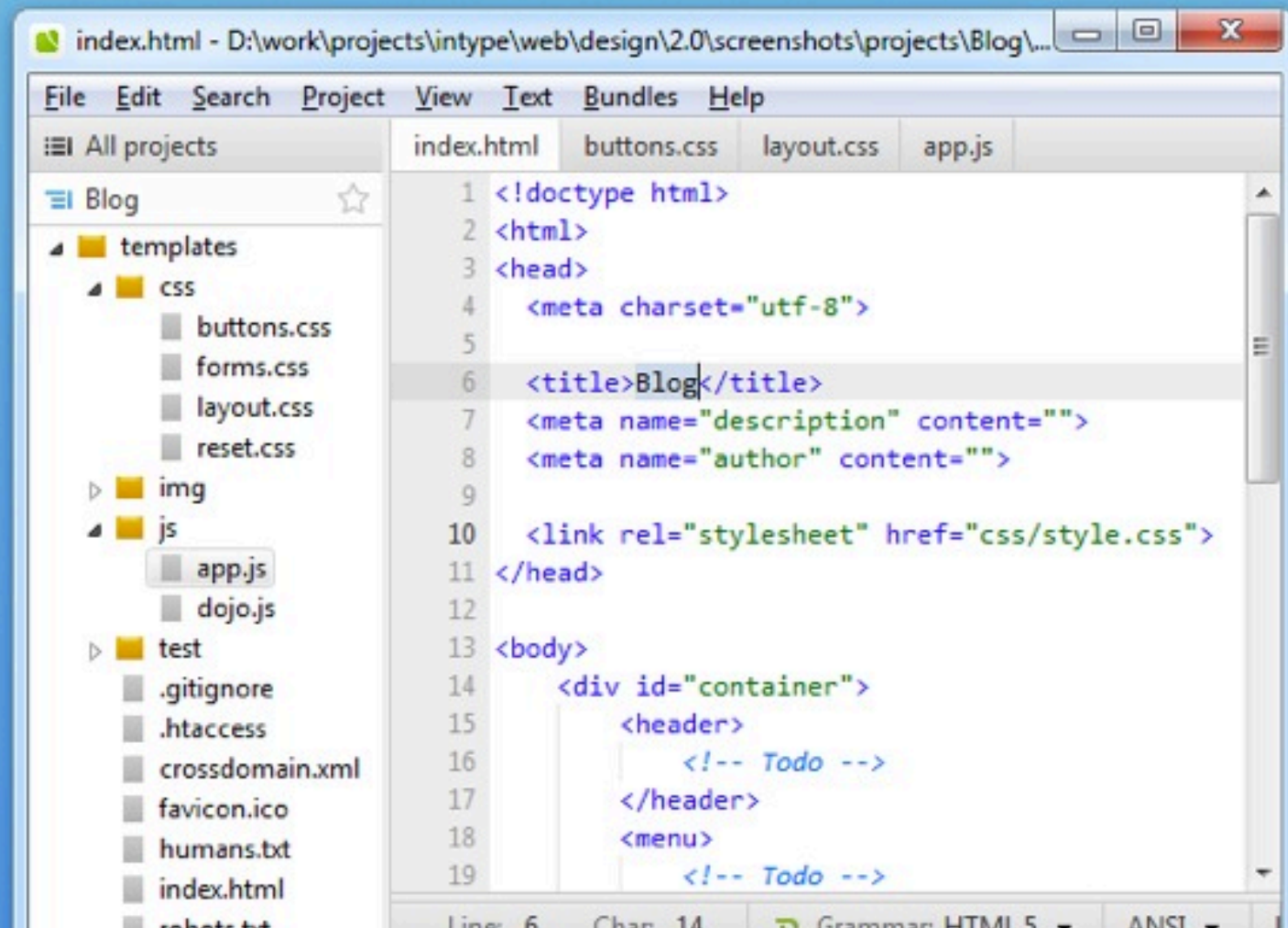
USE WITH CAUTION

It may cause awesomeness  
of your work!

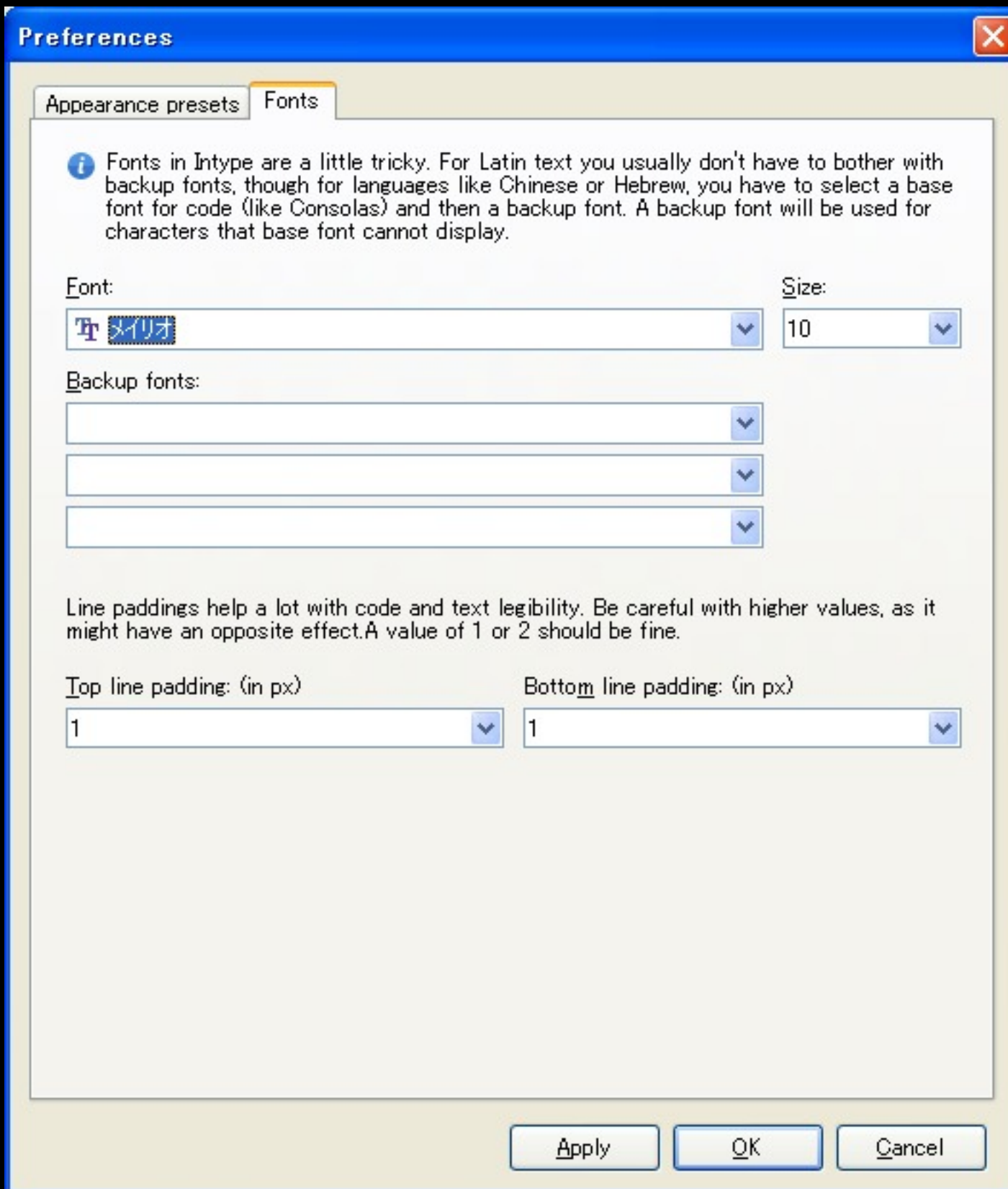
It may cause you to be  
overproductive!

It may cause you to hate all  
other editors!

Download now

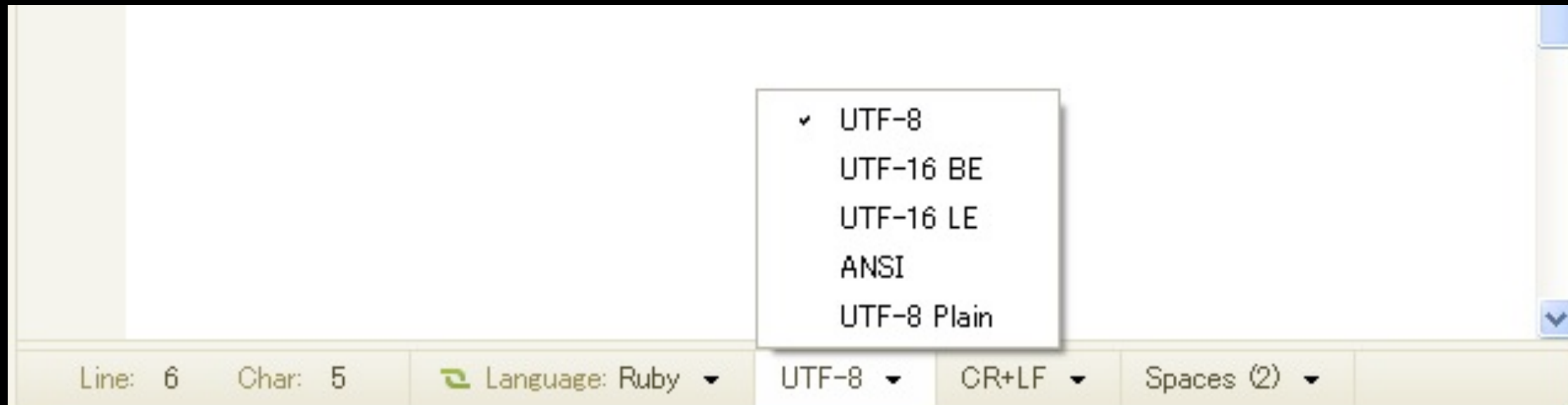


# Windows intype



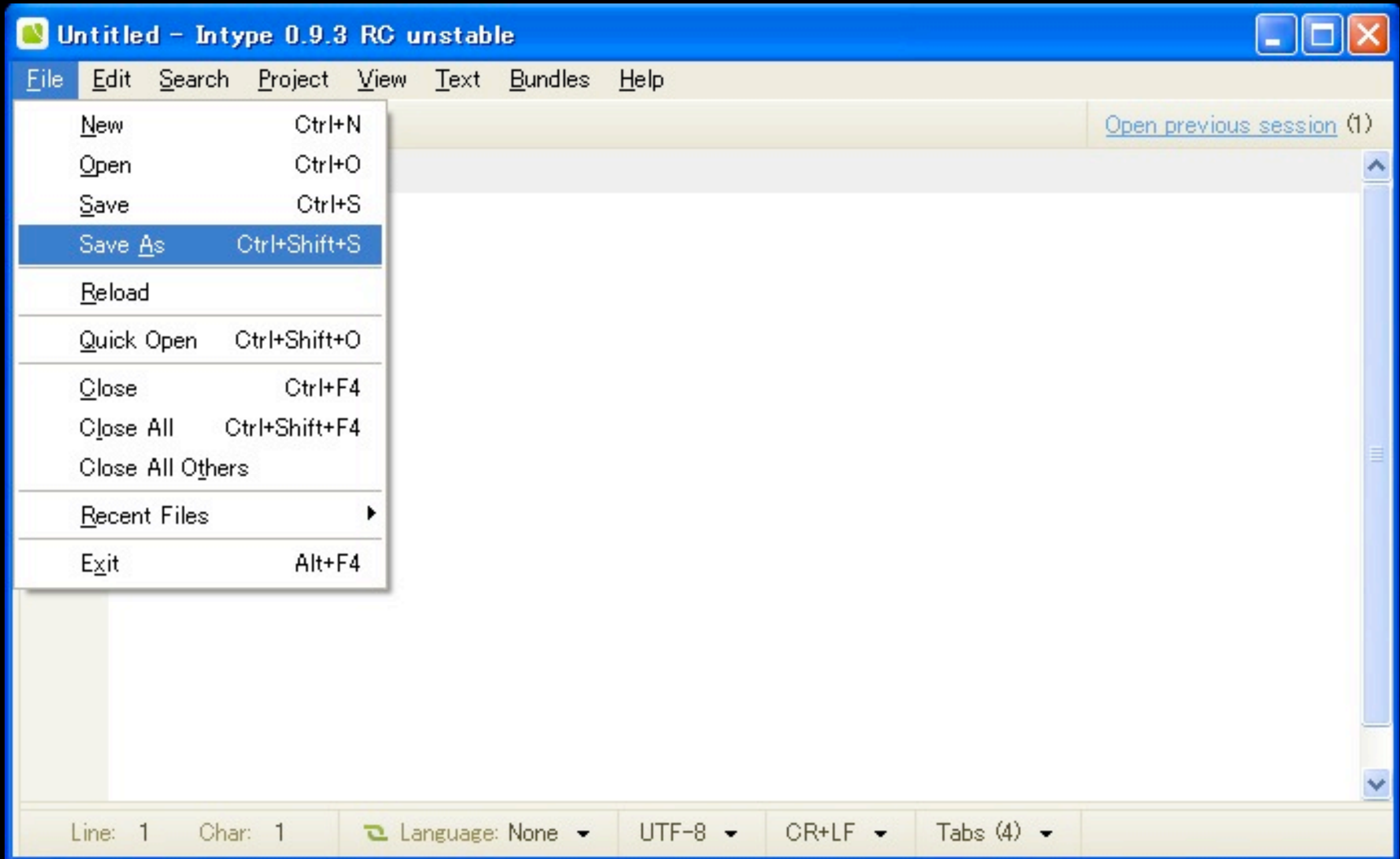
**コードを書くための設定**  
**Edit - Preferences**  
**Fonts タブ**  
**Fontを日本語表示できる**  
**ものに変更（メイリオや**  
**MSゴシック）**

# Windows intype





# Windows intype



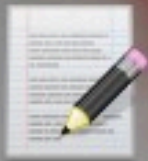
保存時は“Save As” で C:¥Sites¥○○.rbを指定

# Ubuntu

ja 22:01 dev

× テキストエディタ

アプリケーション



テキストエディター

gedit

36

Rubyコードをファイルに記述して実行

helloruby.rbを作って保存

```
print("Hello, Ruby.\n")
```

helloruby.rbを実行

```
$ ruby helloruby.rb  
Hello, Ruby.
```

shellでRubyコードが書かれたファイルを実行するには  
ruby ファイル名

# エラーメッセージ

正しくないプログラムを実行しようとする  
と、エラーメッセージが表示されます。

```
helloruby.rb
```

```
print("Hello, Ruby.\n")
```

```
prin("Hi.") ← 正しくない
```

```
$ ruby helloruby.rb
```

```
Hello, Ruby.
```

```
helloruby.rb:2:in `<main>': undefined method  
`prin' for main:Object (NoMethodError)
```

# エラーメッセージは お得な情報を教えてくれる

Rubyが教えてくれたエラーメッセージ

```
helloruby.rb:2:in `<main>': undefined  
method `prin' for main:Object  
(NoMethodError)
```

日本語訳

helloruby.rb というファイルの 2 行目で  
prinなんてメソッドはないので  
そんなメソッドなかとよエラー が起きたよ

# 日本語と文字コード

## マジックコメント

**chihayafu.rb**

```
# encoding: utf-8  
print("ちはやふるかみよもきかずたつたがわ\n")  
print("からくれないにみずくくるとは\n")
```

```
$ ruby chihayafu.rb
```

# 文字コード

コンピュータで日本語を扱う場合の  
いくつかの表現方法があります。  
これを 文字コード と呼びます。

## 文字コードの種類

UTF-8 : 基本はこれでOK。スタンダードな方法。

CP932(Shift JIS) : Windowsで使われる。

# マジックコメント

そのファイルの文字コードをRubyへ教えるために  
1行目に書くおまじない

```
# encoding: utf-8 ←マジックコメント  
print("ちはやふるかみよもきかずたつたがわ\n")  
print("からくれないにみずくくるとは\n")
```

例)

UTF-8 # encoding: utf-8

Shift JIS # encoding: Shift\_JIS



**メモ**

**VM image**

**<http://bit.ly/rubyvmdata>**

**講義資料第1回**

**<http://bit.ly/ruby-lecture1>**

**講義資料第2回**

**<http://bit.ly/ruby-lecture2>**