PYTHON1: DESARROLLO BACKEND CON PYTHON, POO Y LA ERA DE LAS APLICACIONES WEB

PRINT('ALVARO ZAVALA')
PRINT('ALVARO.ZAVALA-FORMAS@FEDISAL.ORG.SV')

FEDISAL | FORMÁS
CENTRO DE FORMACIÓN AVANZADA PARA PROFESIONALES

INCAF
Instituto Nacional de Capacitación y Formación

PRESENTACIÓN DE PARTICIPANTES

1. Nombre
2. A que se dedica (estudio y/o trabajo)
3. Porque decidió estudiar Python

4. alvarohz@gmail.com

FEDISAL | FORMÁS CENTRO DE FORMACIÓN AVANZADA PARA PROFESIONALES

INCAF Instituto Nacional de Capacitación y Formación

HERRAMIENTA PARA DOCUMENTAR EL CURSO

1. Notion
2. https://www.notion.so/

FEDISAL | FORMÁS CENTRO DE FORMACIÓN AVANZADA PARA PROFESIONALES

INCAF Instituto Nacional de Capacitación y Formación

PYTHON1: DESARROLLO BACKEND CON PYTHON, POO Y LA ERA DE LAS APLICACIONES WEB

1. PROGRAMACION ESTRUCTURADA

FEDISAL | FORMÁS
CENTRO DE FORMACIÓN AVANZADA PARA PROFESIONALES

INCAF
Instituto Nacional de Capacitación y Formación

EVALUACIONES

- Evaluación teórica (20%)
- Evaluación práctica (80%)

FEDISAL | FORMÁS
CENTRO DE FORMACIÓN AVANZADA PARA PROFESIONALES

INCAF
Instituto Nacional de Capacitación y Formación

CONTENIDOS A DESARROLLAR

1. Conceptos fundamentales.
2. Entorno de desarrollo.
3. Variables y tipos de datos.
4. Estructuras de control.
5. Manejo de excepciones.

FUNDAMENTOS DE LOS LENGUAJES DE PROGRAMACIÓN

1. Un lenguaje de programación es un lenguaje que le proporciona a una persona, en este caso el programador, la capacidad de escribir (o programar) una serie de instrucciones o secuencias de órdenes en forma de algoritmos con el fin de controlar el comportamiento físico o lógico de un sistema informático

- Wikipedia

LENGUAJES DE PROGRAMACIÓN DE ALTO NIVEL

1. Alfabeto
2. Léxico
3. Sintaxis
4. Semántica

Un programa escrito en un lenguaje de programación de alto nivel se llama **código fuente**

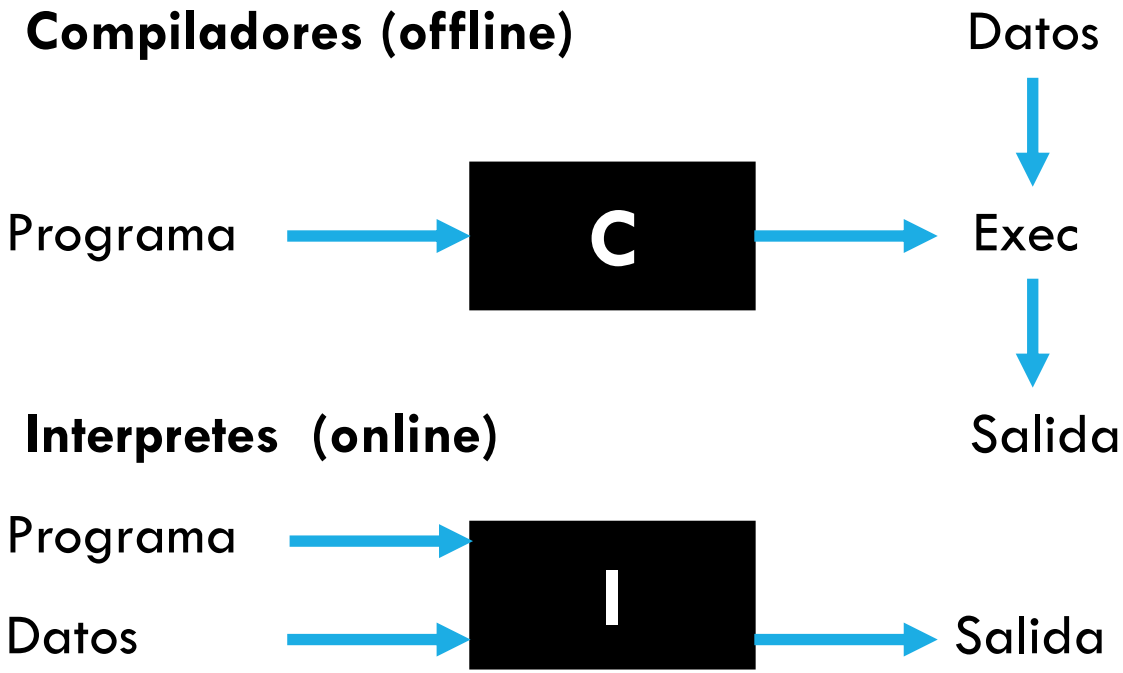
Programa fuente → Traductor → Programa objetivo

COMPILADOS VS INTERPRETADOS

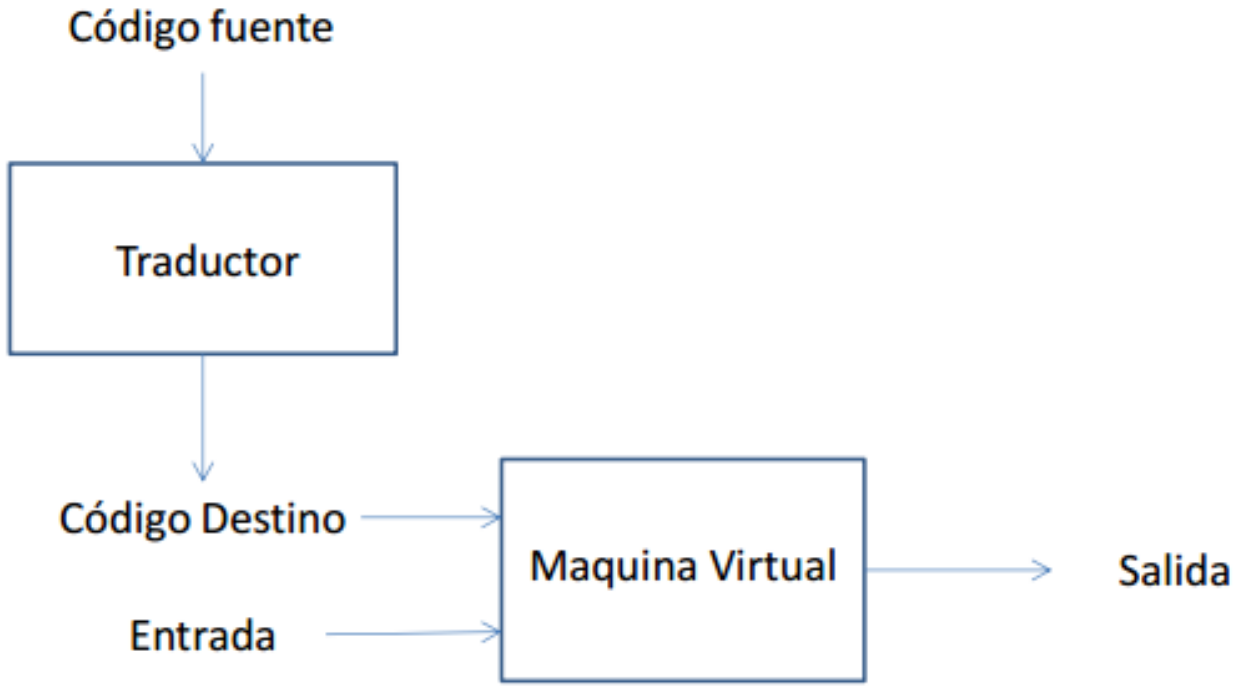
FEDISAL | FORMÁS
CENTRO DE FORMACIÓN AVANZADA PARA PROFESIONALES

INCAF
Instituto Nacional de Capacitación y Formación

COMPILACIÓN VS INTERPRETACIÓN



HÍBRIDOS



Y PYTHON?

- Python es interpretado
- Es gratis
- Python es un lenguaje de programación de alto nivel, interpretado, orientado a objetos y de uso generalizado con semántica dinámica, que se utiliza para la programación de propósito general.

CONCEPTOS FUNDAMENTALES

- Que es Python?
- Python es un lenguaje de programación de alto nivel, multiplataforma y ampliamente documentado, con una curva de aprendizaje muy baja y desarrollos eficiente de proyectos
- Usos de Python

Desarrollo Web

Ciencia y Educación

Desarrollo de Interfaces Gráficos

Desarrollo Software.

Machine Learning

Visualización de Datos

Finanzas y Trading

Ethical Hacking

FEDISAL | FORMÁS CENTRO DE FORMACIÓN AVANZADA PARA PROFESIONALES

INCAF Instituto Nacional de Capacitación y Formación

QUE HACE ESPECIAL A PYTHON

Hay muchas razones. Ya hemos enumerado algunas de ellas, pero vamos a enumerarlas de una manera más práctica:

Es **fácil de aprender** - El tiempo necesario para aprender Python es más corto que en muchos otros lenguajes; esto significa que es posible comenzar la programación real más rápido.

Es **fácil de enseñar** - La carga de trabajo de enseñanza es menor que la que necesitan otros lenguajes; esto significa que el profesor puede poner más énfasis en las técnicas de programación generales (independientes del lenguaje).

Es **fácil de utilizar** - Para escribir software nuevo; a menudo es posible escribir código más rápido cuando se usa Python.

Es **fácil de entender** - A menudo, también es más fácil entender el código de otra persona más rápido si está escrito en Python.

Es **fácil de obtener, instalar y desplegar** - Python es gratuito, abierto y multiplataforma.

PYTHON 2 VS PYTHON 3

- Estas dos versiones de Python no son compatibles entre sí.
- Las secuencias de comandos de Python 2 no se ejecutarán en un entorno de Python 3 y viceversa
- Por lo que si deseas que un intérprete de Python 3 ejecute el código Python 2 anterior, la única solución posible es volver a escribirlo, no desde cero, por supuesto

TIPADO DINÁMICO VS TIPADO ESTÁTICO

- *Tipado*. Es la definición del tipo que tendrá una variable dada.
- El *tipado estático* nos obliga a definir desde el principio el tipo de una variable, ejemplos de lenguajes con tipado estático son C, C++, Pascal, Java, Objetive-C, C#, entre otros.
- El *tipado dinámico* nos da la facilidad de no definir los tipos al declarar una variable, algunos ejemplos son PHP, JavaScript, Grooby, Phyton, Perl, entre otros.

INFERENCIA DE TIPOS

- Un **Sistema de Tipos** de un lenguaje especifica cuales operaciones son válidas para los tipos.
- La inferencia de tipos asigna automáticamente un tipo de datos a una expresión sin necesidad de que el programador lo escriba.
- `(e1: int ^ e2: int) => e1 + e2:Int`

TIPADO FUERTE VS TIPADO DÉBIL (NO TIPADOS)

- Un lenguaje de programación es **fuertemente tipado** si no se permiten violaciones de los tipos de datos, es decir, dado el valor de una variable de un tipo concreto, no se puede usar como si fuera de otro tipo distinto a menos que se haga una conversión.
- Python y Java
- Los lenguajes de programación **no tipados** o **débilmente tipados** no controlan los tipos de las variables que declaran, de este modo, es posible usar variables de cualquier tipo en un mismo escenario. Por ejemplo, una función puede recibir como parámetro un valor entero, cadena de caracteres, flotante, etc.
- **`JavaScript y PHP`**

PYINSTALLER

- Como generamos un ejecutable?

PYINSTALLER

- PyInstaller lee un script de Python escrito. Analiza el código para descubrir todos los demás módulos y bibliotecas que el script necesita para ejecutarse. Luego recoge copias de todos esos archivos - ¡incluyendo el intérprete de Python activo - y los coloca con el script en una sola carpeta, u opcionalmente en un solo archivo ejecutable.
- PyInstaller está probado en Windows, Mac OS X y GNU/Linux. Sin embargo, no es un compilador cruzado: para hacer una aplicación para Windows se ejecuta PyInstaller en Windows; para hacer una aplicación para GNU/Linux se ejecuta en GNU/Linux, etc.

ENTORNO DE DESARROLLO - PYTHON

- Python
- Instalar Python en Windows

  Descargarlo desde https://www.python.org/downloads/

- users/appdata es la carpeta por si no aparece en el path

ENTORNO DE DESARROLLO – SUBLIME TEXT

- Descargar de: https://www.sublimetext.com/
- Poner sublime en modo python3
- crtl + shift + p
- install package -> package control: install package -> python 3
- tools->build system->python3
- crtl + shift + p
- install package -> package control: install package -> all autocomplete
- View->sidebar->show open files
- Crear carpeta en documentos y crear archivo dentro con touch python.py
- open folder->abrir carpeta
- crtl + shift + p
- install package -> package control: install package -> File Icons

ENTORNO DE DESARROLLO - PYCHARM

- Python + PyCharm
- Instalar Python en Windows

  Descargarlo desde https://www.python.org/downloads/

- Instalar Entorno de desarrollo Pycharm
- Existen muchos, como Atom, Sublime Text o Visual Studio Code, pero nosotros usaremos PyCharm.
- https://www.jetbrains.com/es-es/pycharm/download/

ENTORNO DE DESARROLLO – IDLE Y JUPYTER

Además de sus muchos componentes útiles, la instalación estándar de Python 3 contiene una aplicación muy simple pero extremadamente útil llamada IDLE.

IDLE es un acrónimo de: Integrated Development and Learning Environment (Desarrollo Integrado y Entorno de Aprendizaje).

Sin instalación

- Accede a https://jupyter.org/try y busca **"Try JupyterLab"**.

ENTORNO DE DESARROLLO – VISUAL STUDIO CODE

Descarga de https://code.visualstudio.com/

Instalar el Visual Studio Code y el plugin de Microsoft para Python

Y listo.

PRIMER PROGRAMA EN PYTHON

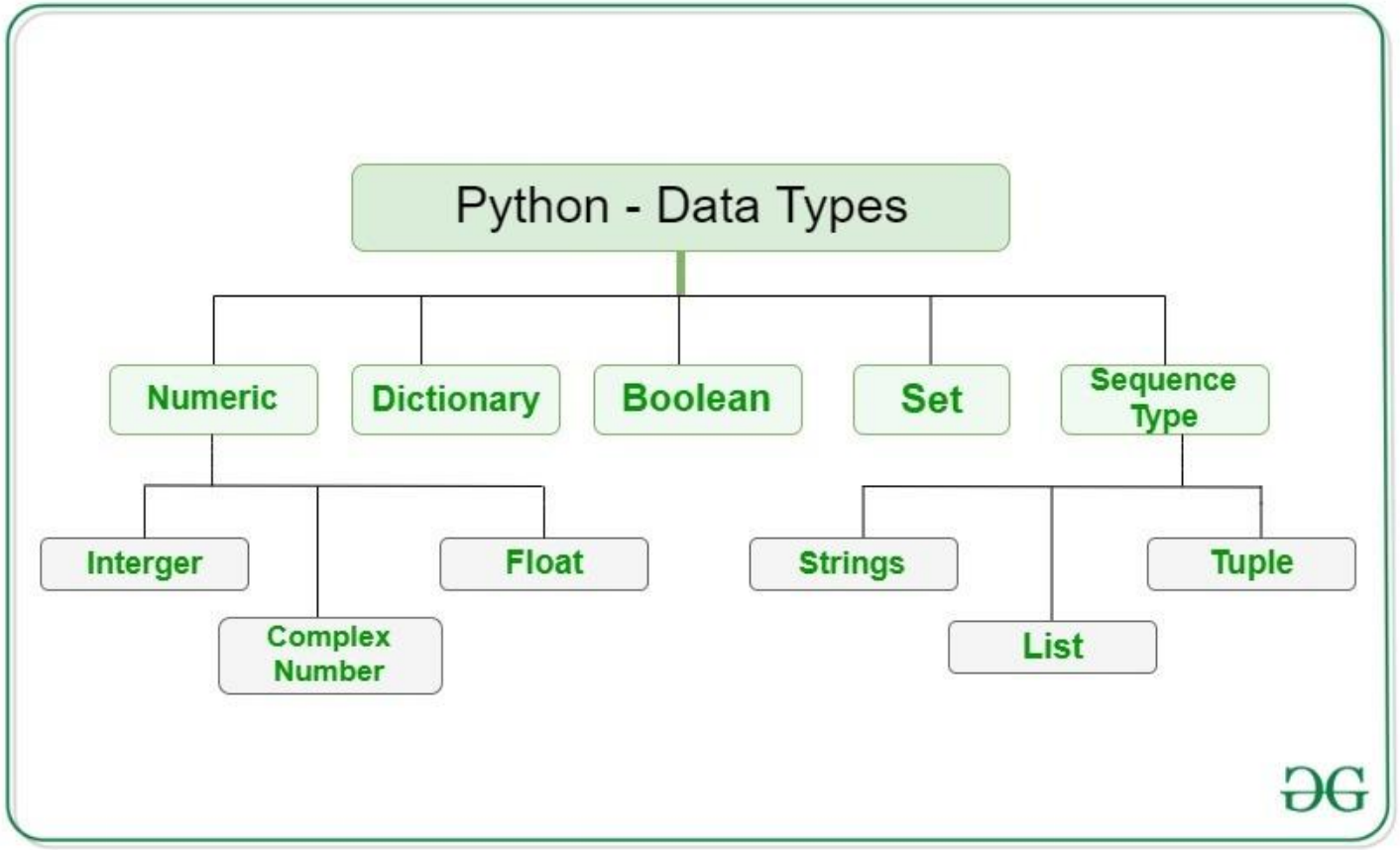
- Seguir indicaciones del instructor
- Definición de variables

- https://www.desuvit.com/11-vscode-keyboard-shortcuts-that-will-boost-your-productivity/

FEDISAL | FORMÁS CENTRO DE FORMACIÓN AVANZADA PARA PROFESIONALES

INCAF Instituto Nacional de Capacitación y Formación

VARIABLES Y TIPOS DE DATOS



OPERADORES ARITMÉTICOS

Arithmetic Operators

Operator	Meaning	Example
+	Addition	4 + 7 ⟶ 11
-	Subtraction	12 - 5 ⟶ 7
*	Multiplication	6 * 6 ⟶ 36
/	Division	30 / 5 ⟶ 6
%	Modulus	10 % 4 ⟶ 2
//	Quotient	18 // 5 ⟶ 3
**	Exponent	3 ** 5 ⟶ 243

FEDISAL | FORMÁS
CENTRO DE FORMACIÓN AVANZADA PARA PROFESIONALES

INCAF
Instituto Nacional de Capacitación y Formación

OPERADORES DE ASIGNACIÓN/REASIGNACIÓN

Operador	Ejemplo	Equivalente
=	x=7	x=7
+=	x+=2	x=x+2 = 7
-=	x-=2	x=x-2 = 5
*=	x*=2	x=x*2 = 14
/=	x/=2	x=x/2 = 3.5
%=	x%=2	x=x%2 = 1
//=	x//=2	x=x//2 = 3
**=	x**=2	x=x**2 = 49

OPERADORES DE COMPARACIÓN

SÍMBOLO	DESCRIPCIÓN	EJEMPLO	BOOLEANO
==	IGUAL QUE	5 == 7	FALSE
!=	DISTINTO QUE	ROJO != VERDE	TRUE
<	MENOR QUE	8 < 12	TRUE
>	MAYOR QUE	12 > 7	TRUE
<=	MENOR O IGUAL QUE	16 <= 17	TRUE
>=	MAYOR O IGUAL QUE	67 >= 72	FALSE

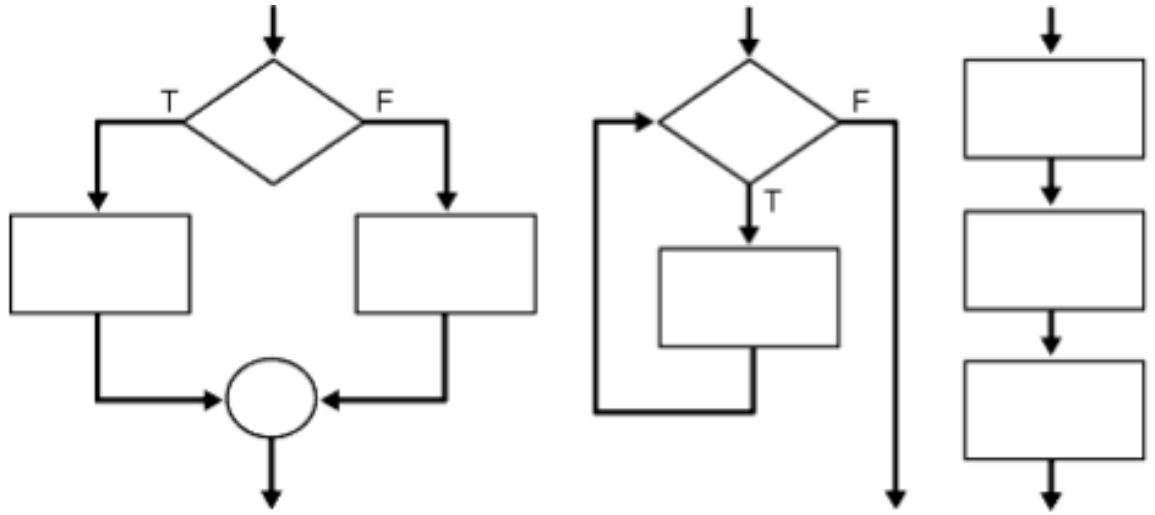
OPERADORES DE LÓGICOS

Operador	Nombre	Ejemplo
and	Devuelve True si ambos elementos son True	True and True = True
or	Devuelve True si al menos un elemento es True	True or False = True
not	Devuelve el contrario, True si es Falso y viceversa	not True = False

ESTRUCTURAS DE CONTROL

- Las estructuras de control, permiten **cambiar el flujo de ejecución de un programa,** haciendo que ciertos bloques de código se ejecuten si y solo si se dan unas condiciones particulares o se repitan una cantidad determinada o indeterminada de veces.

- If
- elif
- Else
- Match (switch)
- For
- while



FEDISAL FORMÁS CENTRO DE FORMACIÓN AVANZADA PARA PROFESIONALES INCAF Instituto Nacional de Capacitación y Formación

DESARROLLO DE EJERCICIOS

Seguir indicaciones del instructor

EXCEPCIONES EN PYTHON

- Las excepciones en Python son una herramienta muy potente que la gran mayoría de lenguajes de programación modernos tienen. Se trata de una forma de controlar el comportamiento de un programa cuando se produce un error.
- Esto es muy importante ya que salvo que tratemos este error, **el programa se parará,** y esto es algo que en determinadas aplicaciones no es una opción válida.

FEDISAL | FORMÁS
CENTRO DE FORMACIÓN AVANZADA PARA PROFESIONALES

INCAF
Instituto Nacional de Capacitación y Formación

EXCEPCIONES EN PYTHON

- imaginemos ahora que por cualquier motivo las variables tienen otro valor, y que por ejemplo b tiene el valor 0. Si intentamos hacer la división entre cero, este programa dará un error y su ejecución terminará de manera abrupta.

```
a = 4; b = 0

print(a/b)

# ZeroDivisionError: division by zero
```

EXCEPCIONES EN PYTHON

- Ese “error” que decimos que ha ocurrido es lanzado por Python (raise en Inglés) ya que la división entre cero es una operación que matemáticamente no está definida.

- Se trata de la excepción ZeroDivisionError. En el siguiente enlace, tenemos un listado de todas las excepciones con las que nos podemos encontrar.

- https://docs.python.org/3/library/exceptions.html

EXCEPCIONE: KEYBOARDINTERRUPT

```
if __name__ == '__main__':
        try:
                main()
        except KeyboardInterrupt:
                sys.exit()
```

USO DE RAISE

- También podemos ser nosotros los que levantemos o lancemos una excepción. Volviendo a los ejemplos usados en el apartado anterior, podemos ser nosotros los que levantemos ZeroDivisionError o NameError usando raise. La sintaxis es muy fácil.

```
if a == b:
        raise Exception("Error", "iguales")
```

PREGUNTAS?

Gracias por su atención!!!!

FEDISAL | FORMÁS CENTRO DE FORMACIÓN AVANZADA PARA PROFESIONALES

INCAF Instituto Nacional de Capacitación y Formación


thank you
merci
gracias
danke
謝謝
ngiyabonga
teşekkür ederim
tapadh leat
go raibh maith agat
obrigado
dziękuję
bedankt
спасибо
mauruuru
hvala
sukriya
kop khun krap
terima kasih
dakujem
arigatô
grazie
takk
dank je
mochchakkeram
merci
рахмат
Баярлалаа
faafetai lava
kiitos
dankie
dhanyavad
nanni
nandri
bayarlalaa
gràçie
enkosi
köszönöm
vinaka
спасибі
blagodaram
mersi
kia ora
barka
welalin
tack
spas
misaotra
matondo
paldies
grazzi
mahalo
djiere dieuf
tau
akun
dankon
ačiū
chnorakaloutioun
gratias ago
gràcies
sulpáy
taiku
sagolun
sobodi
děkuji
mèsi
didi madloba
najis tuke
ありがとう
tanemirt
rahmet
rahmat
kam sah hamnida
তোমাকে ধন্যবাদ
감사합니다
xièxie
ευχαριστώ
diolch
dhanyavadagalu
shukriya
mercé
мерси
trugarez
дякую
mamnun
chokrane
murakoze
хвала
asante
manana
obrigada
tenki