



Reporte entrega 3

En el siguiente reporte se encuentra el registro de todas las funcionalidades que abarca nuestra [página web](#) junto con los requerimientos solicitados en el enunciado.

*Las siguientes descripciones están en sintonía con lo descrito en la rúbrica publicada en Syllabus.

Usuarios y Login (13 pts):

- **Registro de nuevos usuarios (3 pts):** Al acceder a la dirección index, en el caso de **no estar registrado el visitante tendrá la posibilidad de crearse una nueva cuenta** mediante el siguiente formulario:

Crear cuenta

Nombre	Nombre y apellido
Rut	RUT sin puntos con guion
Edad	Edad
Sexo	Prefiero no decir
Dirección	John Street 2805
Clave	Clave

Crear cuenta

Si bien, las BBDD trabajadas solo contaban con los sexos mujer y hombre, agregamos una tercera opción de "Prefiero no decir". Con respecto a la **dirección**, este atributo corresponde a un **drop down que cuenta con todas las direcciones** entregadas por una de las bases de datos de la entrega 2.

En el caso de intentar crear una cuenta con un RUT que ya se encuentra en la BBDD de usuarios, la página arrojará un mensaje que notifica que **ya existe un usuario registrado con ese RUT, dado la opción de volver a registrarse**.

- **Cambio de contraseña (2 pts):** En la página de perfil, el usuario **podrá cambiar su contraseña solo si ingresa de forma correcta la contraseña actual**. De no hacer esto bien, la página arrojará un mensaje notificando que la contraseña anterior es incorrecta, que vuelva intentarlo. Además, no podrá cambiar su contraseña si la "nueva contraseña" se encuentra vacía. Para cambiar la contraseña en la base de datos se llama (en forma de consulta) al procedimiento almacenado cambiar_clave(id_usuario, nueva_clave).
- **Login (3 pts):** En la página index, el usuario tendrá la **opción de iniciar sesión con un rut y contraseña válidos pertenecientes a la BBDD de usuarios**. La columna de contraseña fue creada con un procedimiento almacenado, este procedimiento y su forma de llamarlo se encuentra comentado en la sección de procedimientos almacenados ([ir](#)). Para revisar si el rut y clave coinciden con los de un usuario en específico, se llama al procedimiento almacenado chequear_clave(rut_ingresado, clave_ingresada) la cual retorna un TRUE si es que el rut y clave corresponden a un usuario y FALSE en otro caso. Si el usuario ingresa un rut y/o clave incorrectos o que no coinciden con la base de datos, será redirigido una nueva página en la que se le indicará que cometió un error y le dará la posibilidad de volver al inicio (para volver a intentarlo) o crear una cuenta nueva si lo desea.



clave y/o usuario incorrecto

Para volver a intentarlo debes ir al inicio

[Ir al Inicio](#)

¿No tienes cuenta?

[Crea una cuenta](#)

[Crear](#)

Importación de usuarios (5 pts): La asignación de contraseñas e importación de usuarios se encuentra especificada abajo en la sección de procedimientos almacenados ([ir](#)).

Información personal (11 pts):

- **Información general (3 pts):** En la página de perfil, el usuario podrá ver los siguientes datos:

Hola Momina Read este es tu perfil

ID Usuario	Nombre	RUT	Edad	Sexo
130	Momina Read	62420431-k	37	hombre

Estas son tus direcciones

Direccion	Comuna
Pleasant Street 373	arica
Heather Lane 1081	arica
Route 1 3510	arica
Route 30 4375	arica
Briarwood Drive 3071	arica

- **Historial de compras (3 pts):** En el caso de que el usuario cuente con un historial de compras, podrá consultar su historial y ver los productos comprados ordenados por su ID de compra (por antigüedad):

Hola Poppy Hartman, este es tu historial de compras

ID Compra	ID Producto	Nombre Producto	Cantidad	Precio	ID Tienda	Nombre Tienda
119	181	hamburguesa de verduras	1	5304	73	Almacen Fluido
119	354	pollo congelado	4	8252	73	Almacen Fluido

- **Información específica de jefe de unidad de despachos (5 pts):** En el caso de que el usuario corresponda a un jefe de unidad, al ingresar a su perfil podrá ver la información de los trabajadores correspondientes a su unidad (con el/ella inclusive):

Eres jefe de la unidad 3 a continuación se mostrarán los trabajadores pertenecientes a tu unidad

ID Unidad	Nombre Personal	RUT	Sexo	Edad	Clasificación
3	Tyrese Bishop	54178021-1	mujer	29	administracion
3	Bridie Stanley	15369176-2	hombre	51	ti
3	Haider Hines	24927693-6	hombre	19	ti
3	Jasper Harvey	32359787-1	mujer	54	vendedor

Navegación (15 pts):

- **Página de tiendas (12 pts):**

En esta vista se pueden ver todas las tiendas de la base de datos con su nombre, ID, dirección y comuna. Esto lo hace mediante una query de sql en el archivo tiendas.php. Para hacer consultas a una tienda en específico hay que apretar el botón del ID. También incluye botones en el encabezado de la página para acceder al perfil o cerrar sesión.

Porfavor elija una tienda haciendo click en su id

ID Tienda	Tienda	Direccion	Comuna
0	Almacen Centrales	Franklin Avenue 4858	la ligua
1	Tesoros el dia 42	Laurel Drive 4110	la ligua
2	Casa Antigua	Court Street 2153	la serena
3	Almendras Almacen	Williams Street 3510	santiago
4	Fuera del atico	Orchard Avenue 2495	la serena

- **Vista principal (2 pts):**

Una vez seleccionada la tienda se mostrará esta vista que tiene las indicaciones para realizar cualquiera de las **3 consultas indicadas en el enunciado**, ya sea el top 3 de productos más baratos, búsqueda de algún producto por su nombre o realizar una compra. En esta vista también se ve nuestra funcionalidad adicional que explicaremos más abajo, incluye botones para volver al menú de tiendas, ir al perfil del usuario y cerrar sesión. Esto corresponde al archivo llamado consultas_tienda.php

Consulta top 3 de los productos mas baratos (cualquier sea su tipo)

Top 3

Consulta algun producto por su nombre para ver informacion basica y disponibilidad

texto
 Buscar

Realiza una compra de algun producto ingresando su ID y cantidad

id item
 cantidad
 Comprar

- **Funcionalidad de productos más baratos (5 pts):**

Para hacer esta consulta basta con clickear el botón que dice "top 3", esto redirigirá al usuario a una vista que muestra dos tablas, con los **3 productos más baratos comestibles y los 3 productos más baratos no comestibles** (esto se hace mediante dos consultas de sql en el archivo funcionalidad_1.php). En caso de no haber algún producto de dicha categoría igual se muestra el header de la tabla correspondiente como se indica en el enunciado. Todos los ID de los productos mostrados son clickeables y redirigen al detalle del producto. Además esta vista incluye botones para volver al menú de consultas, ir al perfil del usuario y cerrar sesión.



COMESTIBLES

ID	Nombre	Precio
205	lechuga	5224
300	durazno en conserva	5467
318	pavo	6208

NO COMESTIBLES

ID	Nombre	Precio
278	camiseta	9381
104	chaqueta	9435
236	mouse	11941

- **Funcionalidad de búsqueda de productos (5 pts):**

Para acceder a esta funcionalidad el usuario debe **ingresar algún string** o input de texto en la casilla disponible y luego presionar “Buscar”. En caso de que no ingrese ningún string, se desplegarán todos los **productos disponibles de la tienda que seleccionó**. Todos los ID de los productos mostrados son clickeables y redirigen al detalle del producto. Esto lo hicimos usando consultas de sql en el archivo llamado funcionalidad_2.php. Además esta vista incluye botones para volver al menú de consultas, ir al perfil del usuario y cerrar sesión.

De acuerdo al texto que ingresaste estos son los resultados que encontramos:

ID	Nombre	Precio	Descripcion	Tipo
318	pavo	6208	carne	Comestible
74	pantalla lisa	14478	pantalla de luz	No comestible
321	esparagos frasco	9752	verdura en conserva	Comestible
416	esparagos frasco	6807	verdura en conserva	Comestible
54	papaya en conserva	6280	fruta en conserva	Comestible

- **Funcionalidad de compra de productos:**

Para esta funcionalidad es necesario **escribir el id del producto y la cantidad que deseamos comprar**. Esto nos llevará a una página que nos indicará si es que nuestra compra se hizo efectivamente o si es que hay algún error como por ejemplo, que esta tienda no tenga stock del producto que introducimos, o si es que lo tiene, no tiene cobertura para ninguna de nuestras direcciones. Por otro lado si es que la compra se realiza, esta se agrega a la relación carrito_compras y a la relación compras, chequeando dentro de todas las direcciones del usuario a cual de ellas y hay cobertura de despacho por parte de la tienda..

- **Página de productos (3 pts):**

Desde cualquier página que tenga productos enlistados se puede hacer click en el ID del producto y se desplegará esta vista. En esta se puede **ver el detalle correspondiente al producto seleccionado**. Esto, mediante una consulta de sql en el archivo show_producto.php. Las columnas que se muestran dependen estrictamente de las características del producto. Además, esta vista incluye botones para volver al menú de tiendas, ir al perfil del usuario y cerrar sesión.



Procedimiento Almacenado (20 pts):

Con el objetivo de cumplir los requerimientos planteados en el enunciado definimos los siguientes procedimientos almacenados junto con un código php asociado para poder ejecutarlos en cada caso:

Procedimientos:

1) agregar_columna_clave()

Este procedimiento revisa si es que existe el atributo 'clave' en la relación usuarios, si es que no existe lo agrega y a cada usuario le inserta la clave "cambiar_clave". Es necesario agregar esta columna para poder guardar las claves de los usuarios. Si es que el atributo no existe y fue insertado, entonces este método retorna TRUE, si el atributo ya existía y no fue necesario insertarlo, retorna FALSE.

2) agregar_usuario(nombre, rut, edad, seco, calificacion, id_direccion)

Esta función inserta un nuevo usuario en la relación usuarios.

Comienza chequeando si el rut en cuestión ya se encuentra en la relación usuarios, de no ser así sigue los siguientes pasos:

- La relación usuarios tiene como atributos id, nombre, rut, edad, sexo, clave y calificacion. Para crear un nuevo usuario se necesita un nuevo id que no esté en la tabla, esto lo logramos buscando el máximo id de la relación usuario y le sumamos 1 para obtener un nuevo id. La contraseña que se asigna es 'cambiar_clave'.
- Agrega una dirección al nuevo usuario en la relación direcciones_usuarios la cual tiene como atributos id_usuario e id_direccion. El id de la dirección es el entregado como parámetro y el id del usuario es el mismo id calculado en el párrafo anterior anterior.
- Retorna el id del nuevo usuario

Si el rut ya se encuentra en la relación y su calificación no es 'usuario', entonces le aplica un update solo a su calificación por la de la entrada. La función retorna -2

Si el rut ya se encuentra en la relación, entonces la función retorna -1.

3) cambiar_clave(id_usuario, nueva_clave)

Esta función cambia el valor del atributo 'clave' del usuario con id igual a 'id_usuario' por el valor del parámetro 'nueva_clave'

4) chequear_clave(rut, clave_test)

Esta función chequea si el usuario que tiene el rut de entrada, tiene una clave igual al parámetro 'clave_test', de ser así retorna TRUE. En cualquier otro caso retorna FALSE.

5) chequear_disponibilidad(idtienda, idproducto)

Esta función chequea si es que el producto que introdujimos se vende en la tienda que elegimos comprar, todo esto lo hace a través de sus id. Retorna TRUE si es que la tienda si vende el producto, y FALSE, si es que no lo hace.

6) chequear_despacho(idtienda, comuna)

Esta función chequea si es que la comuna ingresada corresponde a las comunas que cubre el despacho de la tienda.



7) **insertar_compra(idusuario, nombre_comuna, idproducto, cantidad, idtienda)**

Esta función inserta un id de compra, un id de usuario y una dirección a la tabla compras y agrega un id de compra, un id de producto, una cantidad y un id de tienda a la tabla de carrito de compras.

Funcionamiento web (php):

1) Agregar columna

http://codd.ing.puc.cl/~grupo53/procedimientos/agregar_columna.php

Al hacer click en este link, se ejecutarán el procedimiento almacenado **agregar_columna_clave()** como consultas a la base de datos del grupo impar, agregando la columna clave si es que todavía no existe. Analizando el retorno de este procedimiento determina si es que se agregó la columna o no, si es que se agregó se aplica una consulta para obtener los id de todos los usuarios y luego a cada usuario (foreach) se le agrega una clave con el método cambiar **cambiar_clave(id_usuario, nueva_clave)** donde id_usuario corresponde al id del usuario de la iteración y nueva_clave a una clave aleatoria de 8 caracteres generada por la función generateRandomString(8). Como generar numeros aleatorios no es parte de los objetivos de este proyecto esta función, esta se obtuvo a del siguiente link:

<https://stackoverflow.com/questions/4356289/php-random-string-generator>

la primera y última línea definen qué se trata de una función con entrada length con valor predeterminado 10. Luego declara una variable tipo string la cual corresponde a un string con todos los 10 dígitos y caracteres del abecedario inglés en mayúsculas y minúsculas. Luego calcula el largo de este sting para después aplicar un ciclo for desde cero hasta el length. En cada iteración del ciclo concatena un carácter del string seleccionado con un índice aleatorio (entre 0 y el largo -1) a un nuevo string el cual retorna al terminar las iteraciones.

```
#https://stackoverflow.com/questions/4356289/php-random-string-generator
function generateRandomString($length = 10) {
    $characters = '0123456789abcdefghijklmnopqrstuvwxyzABCDEFGHIJKLMNOPQRSTUVWXYZ';
    $charactersLength = strlen($characters);
    $randomString = '';
    for ($i = 0; $i < $length; $i++) {
        $randomString .= $characters[rand(0, $charactersLength - 1)];
    }
    return $randomString;
}
```

2) Agregar admins (importación de usuarios)

http://codd.ing.puc.cl/~grupo53/procedimientos/agregar_admins.php

Al hacer click en este link se aplicará una consulta en la base de datos del grupo 56 para obtener todos los datos de las personas de la relación administrativos, incluyendo la dirección de su unidad de despacho. Luego para cada uno de ellos (foreach) se ejecutará el



procedimiento almacenado en la base de datos del grupo 53 `agregar_usuario()` con parámetros correspondientes a los datos obtenidos de cada persona según corresponda. Recordar que en dirección se asigna la dirección de la unidad. Este procedimiento retorna -1 si es que el usuario ya está en la relación, si es que el usuario no estaba en la relación retorna el id del nuevo usuario el cual se ocupa, en conjunto con la función `generateRandomString(8)`, para consultar le procedimiento almacenado `cambiar_clave(id_usuario, nueva_clave)` y así agregarle una clave random. Este procedimiento adicionalmente muestra, con fines explicativos y pedagógicos, la relación usuario completa (incluyendo todos los id de los usuarios y sus respectivas claves).

3) Otros:

A pesar de que estos procedimientos almacenados fueron diseñados especialmente para lo pedido en el enunciado, también se ocupan para cambiar la clave de los usuarios (en la página perfil) y crear nuevos usuarios (en la página crear cuenta)

Funcionalidad Adicional (5 pts):

- Nuestra Funcionalidad adicional consiste en poder **decidir cuánta cantidad de un cierto producto queremos llevar en nuestra compra**. Esta cantidad se selecciona al mismo tiempo en que elegimos el producto, de esta manera es muy fácil comprar varias unidades de un producto en específico sin tener que hacer compras unitarias de este. Para esta funcionalidad creamos una función a través de procedimientos almacenados llamada `insertar_compra()`, que en la tabla de `carrito_compra()` nos inserta un id de compra, un id de producto, una cantidad y un id de tienda haciendo muy fácil agregar nuevas compras.

USUARIOS Y CONTRASEÑAS:

Disponibles también en [agregar admins](#)

Rut	Clave
18167947-6	JEU717SY
72810804-5	0kCt0WSH
44253237-0	ZQ0M16XD
45026478-4	7JMr066Q
75576764-6	MDq9VXIW
64879120-8	nSiVJbnf
69373271-9	IDyCjvdl
10517073-4	7SF21Rca
98708700-5	VVBbHIRE
58945079-5	Pk2f83kN
75270010-9	PI2Yyea9
98683815-5	8UgZENTZ
72939187-5	xR9NgCUz
20297901-7	ml5VnTjv
77209293-8	0WCi9h5q



18665525-7	A45sr2eu
50062649-6	v2lE2rEb
85973291-7	2kMUfzKP
51153844-0	45kP0iju
54076655-k	n9i3CF18
94821886-0	dB0l9pTe
36552322-3	QuMYiBtl
32366948-1	6wH0tu4x
77477421-1	Dw2E2hu2
13572944-2	r2ElWhmf
79725605-6	G7CTTepr
66183219-3	23VQ6aPh
40675138-4	S6TmzVSP
23000993-7	VzcZCh2y
88198898-4	MjbyLPLV
66199185-2	XVMGvvda
16008678-5	kYmo1o3F
36625090-5	xkllvirZ
87813399-4	FxAuWdVw
93900888-8	L4Ms2quX
30862862-0	LKMdsHh4
43922451-7	SsfGuh7F
95520055-1	umIMRWI1
75016224-k	KGnW4qaz
76591179-6	MUu3hVss
87081187-k	YTJBGX6U
36081017-8	2AUhjcAN
80291272-2	C17UG8zR
37200573-4	fX9D9rpO
56535213-k	vhCQEM3B
77796005-9	GueDEk7c
40350724-5	nCIA01XE
56499633-5	dxxmullY
23526948-1	SWJ7qNMj
86664654-6	13uDg9oo
52916249-9	OPAI0ZU5
62785039-5	7nnvHErj
22145359-k	WAJDwwvL
63951440-4	GaTyaptO
97004364-0	UVnfBD2d
47666047-5	K30RVIBp
88730119-0	TQwpsCsw
53211068-8	2zcChfHn
14046488-0	2Wrnefvm
31706054-8	yig2PL6D
23298212-8	rODuMZJu
57836259-2	lMioORRo



97657967-4	8fKFbS5U
59814228-9	HawzHdqc
89336201-0	XOozQWEu
94432090-3	9D0DZhjA
92170041-5	WwYD66gU
64272663-3	lrax8z2z
91790173-2	98zYiqYy
45467235-6	hRIWakDf
99075946-4	FLvN5RTQ
96392316-3	pMCAtweZ
73342699-3	KMXan1nd
47865441-3	Pez1U40K
88772168-8	swZNljYv
87861767-3	4BtbBiHm
41361801-0	bgQX4LLM
31092780-5	zkgaFS4l
11750224-4	9tVwwd2K
72070561-3	ak5wyrqb
51693674-6	zX7tLSQ7
92365268-k	CgDmUYBE
55776079-2	U8RegTxL
83296811-0	QBKcMdgW
23713134-7	fCpLALrR
70624789-0	8tlqehqL
79425659-4	9bX3JXSw
66258468-1	cKFsVWYp
35662780-6	ejoE7Nxj
40403791-9	HK5SIKZJ
79480331-5	JFwmFd8U
86646016-7	IS8n0Qcn
21666289-k	PNGZKBA5
52703981-9	TjCGe2VF
83603309-4	IM2aAfz4
47326835-3	zOJI0liJ
49337371-4	ETXjxURB
87442136-7	qRQOGuwH
48679409-7	oDWIIitsT
36259502-9	1qdHWqOo
29646441-4	GogaJeWc
33402225-0	HnXTkcYI
95855049-9	17QVdL4c
51755919-9	YUWI433L
49993665-6	8PHW9pTX
69141385-3	4DibwTZr
33481996-5	tWNQHdel
68505645-3	d91A8JIn
47067447-4	iurpKGoZ



58557550-k	KD8d6j04
27650119-4	JnvZvOuU
18812547-6	X4S8vwXh
66625711-1	PYMN7lfq
52189199-8	Mdn1td7F
28012525-3	yTNGcPv9
48795285-0	aI8hG8V6
89321752-5	pwHQxL11
35066595-1	NI1mnGj5
42621036-3	WLB97Dro
66954467-7	uURlrztr
18446690-2	DiaI5bk1
65227086-7	Je8rhS8s
35661237-k	uNAwhOY5
67913619-4	vWJexO3U
83290634-4	fpGMhr2X
62014673-0	Afr5j1Ce
92311983-3	lfcj3KkG
59517619-0	AeN61T9O
42655208-6	U6FU15Ho
65573855-k	xsM2guro
62420431-k	BrES28ns
56064569-4	24RJe6fR
94422835-7	xFSDACyl
35668656-k	rAhvAjQh
95437335-5	1uGfVA3g
63532763-4	Bh6fRDar
46680387-1	tE0EYPv3
85297873-2	AbsaWARR
88132030-4	3VNxij9V
39473167-6	NNEd7UI0
58154631-9	m81YU0IK
38830395-6	8lwUxVCj
36030482-5	57ZIW7YS
95667888-9	tXacSChp
97996738-1	9tEmVEbs
96007294-4	vU9hzTip
52176953-k	TElarXJu
59332921-6	Ht44q5zN
23298314-0	C3JAHQ7M
31113786-7	EgBZhVuL
87991170-2	pS479HEQ
87631642-0	0r2gbbuF
17509928-k	9gVj3SdN
53089483-5	WcdjJW9J
75523496-6	cJPv9fxq
87734759-1	W17292or



34424260-7	uE3G83QQ
66920820-0	UISgm4QC
93057304-3	1vREfhLO
82673938-k	hjS7hmuK
14920531-4	iHtRkXw9
96260856-6	u3MfW95n
86103635-9	N3Lv65JX
94384917-k	zEAfvvmxc
47329036-7	nh3F8rR8
44867899-7	bvmkofKI
32690774-k	e06gSWZF
10544563-6	F8A82awP
28485302-4	kwTirgJw
63214394-k	DaDmlBrw
36044904-1	n7Zn1h5M
10206801-7	tDtqm98G
34566442-4	RcWFTQeY
65238271-1	02i9XWu4
54230570-3	aAKGwPDA
76871405-3	0dUyUxM0
38204359-6	gjaYg0Nc
33456182-8	YfdtDyOt
75806614-2	DYjpQeSL
19844189-9	BfBhkBBE
70735043-1	EdEUoBJK
64221026-2	NAd0f5x8
50453382-4	uzYbWao6
87649058-7	E6HA659j
75153238-5	4NqDHUKW
12348490-8	FoPgbvVM
92445451-2	zxPsZYhH
49604231-k	A9DmbR8N
35288330-1	AY5Gmf5f
90870389-8	irZFqZ8h
52157315-5	TGSK242f
35820841-k	Us2lOqKh
66361406-1	FcVq6ruu
20843742-9	dfyWk49u
78059417-9	gA1pdzlF
98417733-k	jZ0CxoNJ
59170372-2	kbXsHG14
25070905-6	IVPKukh4
60370162-3	z1HMp1FN
54982068-9	YmR5Hfs5
66336340-9	gDB1uk7e
35685353-9	R59hh03c
97102977-3	CQzzan8D



58155861-9	QameJs4b
42168093-0	dJvWU1gP
59885368-1	Sad6gcnd
61554580-5	0mw1gUj5
66646618-7	jFvg0W9Q
15298663-7	13NR59Uy
62778278-0	AyhP3nm1
38287473-0	ZeZhpC03
11644968-4	kXJvvO3H
97781705-6	SSHmjE3V
80611905-9	aNdFmbQk
97343582-5	uqwOgZSt
79693636-3	W9eAedaP
69140468-4	8v516JUt
48925896-k	Mw93N0dV
20640357-8	rkLm0f0X
33016426-3	uZaZ60iL
56790702-3	jYiAqssq7
45276568-3	mQ24TiSU
20335324-3	fuakPJhM
42226798-0	fQYxEnLO
43129309-9	Ewk8N83Z
26905798-k	WOUDUzR1
91119340-k	tV7PNKgo
77625718-4	DCsDOTKT
81698647-8	AcB2h29X
15333675-k	pFk78GaY
61993529-2	DLig3jAJ
22060310-5	wnQ77Wdg
77520112-6	uh56ABnJ
75346943-5	5J98h58G
62550748-0	DnaHJJ29
77814675-4	qc6w6IYV
64370666-0	CxWO1RIJ
28752611-3	RZ8NWAZQ
74644456-7	rkPegqoL
37500745-2	s1aXsc3V
56119224-3	2Wvw3UJt
59919311-1	FkeWyN4s
11087572-k	B0sgY90U
12033288-0	mGc0nzVr
49891081-5	b3lpm3ZS
55705480-4	6q6M8rzD
98642893-3	hf8RMGkK
49255740-4	0410zsks8
63360085-6	vDGReEe2
26257768-6	25hROC1T



17474518-8	nz8HFNai
65884381-8	zsmIWEav
47287276-1	knlfqFvu
33078552-7	Xl0wwbKb
48776848-0	l1eWZaxj
16222268-6	oksZYdqv
95226644-6	tImJMe4x
17723136-3	iwWc5Evj
83584289-4	Fb5PfHOc
92554015-3	KVdaqPOD
48345899-1	HbEzxi6l
49036882-5	BFAVitir
96964691-9	d5X5h2iG
16775393-0	NRIAeKbe
51242629-8	7sezjL9P
88806288-2	wQR2frPM
57289805-9	6c3lJofc
49860637-7	gNbl5w2f
27844046-k	u59QspPq
45504619-k	SarxDIDP
73453102-2	Nb0OKI7k
84569598-9	WZaMNduN
94688326-3	TKUCnYbh
84078768-0	9N4mG3J9
91223172-0	HyO2Rfx0
51317447-0	54SbjPHu
78463527-9	VUEFiGU
46025545-7	zskDkhWm
40424433-7	SScZbw2Y
68428477-0	VqbdHhGQ
28630079-0	IJMxFJKm
63824003-3	cC0LOqT8
58307021-4	peJp9dhR
77983700-9	43iL4wfJ
57189229-4	Tsc6Zhvn
41581842-4	VTroQTfl
55106753-k	xmdJS9YE
19735595-6	hQeRcCil
45024924-6	GHuMWYon
63884576-8	xdvHCYKd
61262323-6	vilEDa6x
24129722-5	mwVIIxxY
39567466-8	ZBK7qyaU
94888169-1	s7SMIkRM
31642814-2	5ch0lb46
79195789-3	LdEWUoc2
56292778-6	JolceF9C



33670116-3	o9AzVOHc
48040287-1	BhYHLX3o
14761964-2	b05bLnhb
27712443-2	hMvSf3I9
51853396-7	pSLpaN20
23711490-6	vDSWBbal
53523140-0	4iNIFSRD
49974632-6	48GafMH1
95598580-k	TeiBbwRo
49798131-k	M3jKToCo
71591727-0	xtnhdylZ
30760014-5	rT1XvPH9
33904341-8	SnFCIkU6
78331019-8	uTjOvoPI
64903395-1	RZmT4gTT
45125749-8	vzgOD4NO
81498697-7	i8kZ6Exg
78599189-3	LsjHWH7M
85682952-9	hrOXdjjn
47738259-2	1jQn9mPH
48425773-6	Ng1khzeB
82645956-5	hhzZe62x
55956975-5	kv8IrMcw
99875339-2	nOvhZ2c9
29858450-6	5NBwXorf
74759859-2	kDUkUHyF
46012714-9	5FmjGlks
43630930-9	XPqa6Tt8
56425505-k	xNh72qHM
16239796-6	pitSBVDW
10165029-4	c81K6KSR
82809947-7	LY7sqtou
37492390-0	1P1IWTPX
87544765-3	GezSNjWG
97901764-2	coJrZaHB
99294258-4	5zj8XvQF
48060937-9	X4A0fXc5
85411178-7	yiAZkMhz
10959343-5	Dn7qwYwc
74630833-7	a89qCkhF
20909860-1	LWM2Alix
23748548-3	SdANF2W7
25572424-k	bAOuoAlj
75653640-0	mDA0sXCZ
49359132-0	8Mo5Qslp
50270274-2	83gm6hmt
46080300-4	jWqA7oVk



99265623-9	cNCjHQ5k
75039658-5	m7jyptf0
79735673-5	thDBSqW9
34864642-7	cZUFO4VX
38563658-k	fhBtRQix
13463271-2	DTvZPGXb
50109588-5	UtTizEKK
44126502-6	3EJXSqJ6
73352137-6	lcbsRF9P
70156418-9	Q3Jr65NV
96557795-5	zg4GD9oa
76413818-k	et3z5DgY
66119831-1	BStTDyCZ
81600879-4	zXQ3qo6J
69698793-9	YmKCnIJJ
91888069-0	AM02Gi5B
73075290-3	DILrBUVi
95932617-7	rnDN3wD3
69402733-4	41hLubto
67391164-1	KJHLVAdM
81492646-k	hlkGRDqV
123	123
37630974-6	K3drtfUA
45631604-2	UjQRDvUO
29728267-0	TTDmaQ2V
61662429-6	jap2dOuS
63688364-6	8cBR3n9I
95185947-8	PsAPMPS5
54106148-7	mq5dEE2o
54178021-1	gASfUvgm
32359787-1	ARrwp5WY
35776394-0	7HEoSsRZ
44171043-7	VSce9Bzj
61514910-1	pYOWwUiJ
59089616-0	bzKyvVd8
40588157-8	4xWyWhl0
62820832-8	K47glwLr
61995722-9	Ikctv1dL
60211995-5	MAGPpeno
74357811-2	Ou80iaiq
15614052-k	v0YLPq6c
27852199-0	jwwwp0op
74895468-6	dUPRem2X
45717384-9	UIXI9VsB
22274441-5	dfdozxuo
25342278-5	M940fKTI
65855526-k	3rGLW70g



31690607-9	z0lblicD
13938471-7	d0dd7dBg
12957899-8	6tNMT5rS
46187096-1	LYITDbAI
79446327-1	MHThPJtr
18969557-8	ivp27XdT
21840388-3	sl2sur3G
47673879-2	4yLFo5QP
33461543-k	OIP7qz5u
69705129-5	YLIWGRYt
43776217-1	K0c8zQk0
59445067-1	d5dUDloM
77941465-5	h3EoorR8
60517573-2	vvQDmhOY
46491043-3	IWXJike1
38405094-8	T3XkAUjp
79625227-8	vJyLXovN
34352032-8	MFXD7I87
58748811-6	bbdRTCAy
62832139-6	x5yuWJgn
11091024-k	PlgJc2N6
29676675-5	M4ItySh2
29989393-6	kg05zBvr
97632703-9	ZXdTxSo
34882918-1	aKTsmIac
49166815-6	OxQnfiDy
30854312-9	dlo50EyA
60987426-0	LUW4L8RI
11554909-k	IZEKI2Fd
91059844-9	qFKw07ZE
32628385-1	RBNNRi2O
25368817-3	13XtO2jE
99748016-3	mRvh1B2D
22479749-4	w4xYyufv
80843691-4	MhUJgWnc
28738037-2	sVjrqhTP
20032612-1	grNhu1NO
10669832-5	nOhKS106
22532060-8	e3SB0uaN
17459269-1	5LkDvABW
48433213-4	UGqvVF9
35785969-7	AZHWSu5f
60854678-2	F6iQ6Cwj
23970806-4	zKH1OuXR
23648509-9	XsNKebLo
58303600-8	HCvw6WxL
23819538-1	i1e9w6ty



86164302-6	7FmFERcc
70116705-8	twwTxTNd
30928102-0	AKqKLaVx
33466037-0	gV4fpS0J
63483183-5	i4b23Jiw
88697741-7	vxmrFTXI
67414624-8	mQlxBCeG
85122660-5	8zbzKfGr
56485501-4	vGBI9UEf
77214559-4	toxm0QV5
92898031-6	6EV6Fwn3
58844715-4	lex2kij2
80698616-k	6QS Ny6Jl
40127235-6	EI3br2ZS
51433500-1	rzGWqqQj
58976828-0	Gh55lo09
71162268-3	gZuaHbRA
82319208-8	OXHhLioU
89265324-0	LA vABXYC
37981852-8	uZ3Em342
53783913-9	twKu421f
45751112-4	4x4k68Qe
11798235-1	aXRnoJBs
41091494-8	8csyhIPS
12166206-k	LDzzFQOr
11557739-5	L3Hj0XIH
54475883-7	vlNacXPm
70012917-9	3q25Z0rE
87859472-k	bA0e4gps
30364676-0	rXy8FZ10
82376572-k	XGnwqUi y
13068977-9	GEegc8vf
47981838-k	yIOH0lni
29950437-9	UoQiPtgD
14240870-8	ceD3wZB3
59150917-9	obwmKKHK
60158787-4	NmInIDRI
20672410-2	o7bRBj1i
42679613-9	FUJOYuYp
17286061-3	Lmbji5HT
58963604-k	FwZ4wBgD
85778387-5	m0gzSN6Q
88104231-2	t11U2thH
33874298-3	42u6N3Tz
18902520-3	SNi3ynCW
78459862-4	WNFDtNna
26113418-7	RdlHb8tT



86339917-3	mvUX3ig8
81203114-7	SNhkG4AM
38072383-2	RqHaG2F3
22150901-3	weZyGkT6
83640016-k	ijy09edH
77354291-0	NimobVH3
36706688-1	EWcTo27m
15369176-2	qidb4I18
27143992-k	P6nqckjL
19410523-1	B22ybjgg
44372095-2	lz7edRRS
17159919-9	wvzKRThV
44822766-9	jqDlxYH
90739234-1	3wVEvuuC
81560843-7	gHt0qtYO
22566169-3	bsrPoPdA
76563311-7	0n2J2HgK
75910499-4	sbJWEsnq
85055980-5	QiYyAhvK
57931953-4	2JTk2HIF
38235761-2	0dtuUN9B
65731455-2	aEd2yUOT
20789297-1	xiGelgt0
25497638-5	l8Ix86EO
82993651-8	z0Rayljo
44380500-1	LvTFaE0b
50319391-4	JNORDLQn
81602032-8	UhlypAWE
37839960-2	CjlOB1Sj
31845287-3	HI7cC2zN
15451762-6	Q2o0alfp
46188032-0	xoOoeWt8
46033111-0	JfqGCUR2
83029887-8	JlpzxCS
95759065-9	or7JgE9v
25327705-k	XMN7EzwJ
82693313-5	ITsN8ZnT
64501886-9	MkakFzkp
54593722-0	iscBLmnS
16689945-1	KnvDehJR
92951996-5	Ex2bmjQo
71180222-3	Q8XDvgvk
54245196-3	YKZe4oBI
52650167-5	BNxrfOHh
67314332-6	kjzLco4X
28370900-0	AWglbcWN
89213188-0	d0JaZEZd



34823992-9	sGsJSYVz
66284985-5	ITAJKI17
34975310-3	OXPaLGv1
18067050-5	NA8JeYWa
89118524-3	PmPj85EH
22783728-4	txuX7dsj
60038263-2	mhnrrjcl2
72611524-9	ydpY8ZS0
14309109-0	Dom0RS4f
13370165-6	9MifgvLL
73418283-4	CximXhp0
59060940-4	UCQQ09r3
67170407-k	1c2SQMqn
97754557-9	RuemlbVg
34996262-4	mdrhED7i
36600416-5	UUuYhFlO
81168161-k	tVF3W05M
19533896-5	TUqmlsPR
25309713-2	G1NW9WEg
64268937-1	cTjVjJse
70202613-k	djDSi2E5
48630167-8	Ufp4dD1x
65809271-5	NixQQEc5
35806868-5	g0KYkCc5
96332059-0	WMaLlgo1
97166191-7	hawEDJ3s
97985609-1	26FOKdJW
45644100-9	LPTzi5ov
23494123-2	XgsQFeRP
65175008-3	HrvnZBtJ
22825105-4	Cel5VivL
36918308-7	7Y21H8S1
88726486-4	o6qQR5sg
44973013-5	wykUt10q
58038177-4	0FstC4az
83605517-9	hvc9iB9e
54363208-2	w3C3x9HX
11784418-8	lqxnf230
54480341-7	sW7xad83
74748755-3	Fn4JdGY1
42814101-6	NMZehjkN
12885071-6	EuxCXo10
88151779-5	qyUAkWS1
81783289-k	70Az9R70
28273728-0	zXi1Ba5J
97355372-0	mrHWwyMX
52778909-5	RqDeCSRb



27161682-1	0DLyticEf
99347364-2	Blw6tzqQ
59129614-0	nIqsRHB
34349789-k	T8tkhoiZ
20774234-1	I2CoZuJX
89671041-9	DHrgidSS
17341224-k	sbaqf5Ud
48333719-1	ZNI3Yg3n
58812264-6	OAVdS4zC
26240285-1	bcqnFXN2
29381516-k	yDlwGSxr
60926655-4	k5QlXSUs
88413708-k	latBv4vg
84891139-9	AiEBq1pO
14156767-5	bOQBIRxm
11904025-6	WHgwp84
16943787-4	y2eSQHzV
59537115-5	VxeXshH5
10927357-0	UP5hXKxO
63216586-2	Ls46Jex1
54676049-9	rY0Upazu
47121914-2	o4gHFxxR
73026642-1	BTBWTLlb
34414978-k	SKYhyL97
21016824-9	lyU1d0Ad
59749734-2	nloKu3HA
70433997-6	uqaZM6Cb
48553208-0	00I6cUKU
73454838-3	PIWHOwMZ
73842540-5	T3Xsh37C
58648727-2	0GpeKBS7
95656123-k	Naz0fsbO
81428035-7	JwWbDBbn
46805723-9	fGWsUFUG
96371883-7	ecHYiamQ
17611254-9	DvBefahA
56197957-k	jz2r8wTk
54546935-9	5caxst7Y
48269008-4	c8LODXK5
50942616-3	CmLIBmj
84327091-3	Lqhxeb4B
48007583-8	O2jPXWu8
14585428-8	QjFKLmYO
25450277-4	Eyz4IxJE
95199785-4	aT4JEKMv
13947632-8	TCdAlcdr
16697747-9	lyUJnzlm



37285914-8	sF9z9wL2
11118739-8	EewDITXD
33461512-k	cAfKPwcC
12920124-k	VyuiFOtA
15955355-8	oanvguKb
52792841-9	BQdgckYK
82533150-6	5v0e82iE
77484278-0	Ir4Q2qxp
52258791-5	NAw4ywVK
24927693-6	T4daDTMs
25696685-9	EvKo2lyU
18984900-1	YZE5tmU1
97163657-2	LkwgdRYb
42714709-6	FxRNd76g
50814284-6	XOjInclq
a	uKWJvBs3
1	KFIYlyY0
111	2BdgllB5
19433177-0	R6IEOlgQ
1234	qDcxydqq
741-1	123