

JAVASCRIPT → PARTE 1 → DICHIARAZIONE E ASSEGNAZIONE CC BY-NC-SA

ESERCIZIO

Creare un progetto Javascript che dichiari e contestualmente inizializzi la costante **X** a 10; che dichiari la variabile **y** e successivamente assegni **X** a **y**; che dichiari e contestualmente inizializzi la variabile **z** al prodotto di **X** per **y** diviso 2.

11-01-01-01

VERSIONE 1.3 - DIAPOSITIVA 2ALESSANDRO URSOMANDO

ESERCIZIO

Creare un progetto Javascript che dichiari e contestualmente inicializzi la costante **MAX** a 15; che dichiari le variabili **a**, **b** e **c** contemporaneamente; che assegni ad **a**, **b** e **c** rispettivamente i numeri 10, 11 e 12; che successivamente incrementi le tre variabili della costante dichiarata in apertura e che infine vi sottragga un valore numerico costante a scelta del programmatore (senza dichiarare la costante).

11-01-01-02

ESERCIZIO

Creare un progetto Javascript che dichiari e contestualmente inicializzi la variabile **v1** a 10; che dichiari e contestualmente inicializzi la costante **C1** alla stringa "Alessandro"; che successivamente assegni a **v1** la concatenazione delle due strutture dati nell'ordine in cui sono state dichiarate mediante l'operatore di autoassegnamento. Successivamente si crei una variabile **v2** che valga **true** se e solo se **v1** vale "Alessandro10".

11-01-02-01

ESERCIZI

Creare un progetto Javascript che dichiari e contestualmente inizializzi la costante **MIN** a 10 e la costante **MAX** a 99; che dichiari e successivamente assegni alla variabile **dueCifre** un valore a caso tra i due valori.

11-01-03-01

ESERCIZI

Creare un progetto Javascript che inserisca la costante numerica 1000 in una variabile **x**; che poi aggiunga il valore della costante matematica **e** (numero di Eulero) ad **x**; e che infine mostri la radice quadrata della metà di **x** in un popup.

11-01-04-01

Creare un progetto Javascript che mostri lo stesso output dell'esercizio precedente arrotondato alla seconda cifra decimale.

11-01-04-02

Creare un progetto Javascript che mostri in un popup un numero casuale palindromo di 5 cifre.

11-01-04-03

Creare un progetto Javascript che mostri in un popup un numero pari di 3 cifre.

11-01-04-04

Creare un progetto Javascript che mostri in un popup un orario nella forma hh:mm o h:mm.

11-01-04-05

ESERCIZI

Creare un progetto Javascript che mostri una pagina WEB con la scritta "Ciao Mondo" di dimensione casuale compresa tra i 30 e i 40 punti (si usino dei CSS in linea).

11-01-04-06

Creare un progetto Javascript che crei un numero casuale compreso tra 100 e 1000 e che mostri una pagina WEB che riporti il valore del logaritmo in base due di quel numero (arrotondato alla seconda cifra).

11-01-04-08

Creare un progetto Javascript che crei un numero casuale compreso tra -5 e 5; che calcoli la potenza di 2 per il numero creato e che mostri una pagina WEB con un messaggio del tipo: "2 elevato a 0 è 1".

11-01-04-07

Creare un progetto Javascript che mostri una pagina WEB che riporti un importo minore di 100 euro casuale (due cifre decimali).

11-01-04-09

ESERCIZI

Creare un progetto Javascript che dichiari e contestualmente inicializzi la variabile **x1** al valore numerico 10; che dichiari e contestualmente inicializzi la variabile **x2** alla costante pi greco; che dichiari e contestualmente inicializzi la variabile **x3** al risultato dell'espressione **x1 + x2**.

Il progetto concluda presentando un popup con il valore contenuto in **x3** arrotondato e realizzi una pagina WEB che presenti un paragrafo il cui contenuto sarà il tipo della variabile **x3**. (Gli output attesi sono 13 e number.)

11-01-05-01

Creare un progetto Javascript che dichiari e contestualmente inicializzi la variabile **x1** al valore stringa "10"; che dichiari e contestualmente inicializzi la variabile **x2** alla costante pi greco; che dichiari e contestualmente inicializzi la variabile **x3** al risultato dell'espressione **x1 + x2**.

Il progetto concluda presentando un popup con il valore contenuto in **x3** arrotondato e realizzi una pagina WEB che presenti un paragrafo il cui contenuto sarà il tipo della variabile **x3**. (Gli output attesi sono 103 e string.)

11-01-05-02

ESERCIZI

Creare un progetto Javascript che dichiari e contestualmente inizializzi la variabile **mese** ad un numero casuale tra 1 e 12; che dichiari e contestualmente inizializzi la variabile **giorno** ad un numero casuale tra 1 e 30 e che produca una pagina WEB che mostri in output la data prodotta.

Infine, se la data corrisponde al 19 Aprile, il progetto stampi a video: "è il compleanno del prof".

11-02-01-01

ESERCIZI

Creare un progetto Javascript che produca una data a caso (solo mese e giorno, considerando ogni mese di 30 giorni) e che mostri un popup con la scritta "Siamo nella prima metà dell'anno" se la quantità di giorni trascorsi dal primo dell'anno alla data prodotta è minore di 180 o la scritta "Siamo nella seconda metà dell'anno" nell'altro caso.

11-02-02-01

ESERCIZI

Creare un progetto Javascript che produca un numero casuale tra 1 e 12 e che comunichi mediante popup quanti giorni ci sono nel mese corrispondente (si consideri Febbraio di 28 giorni).

11-02-03-01

ESERCIZI

Creare un progetto Javascript che produca e mostri mediante popop un orario casuale valido. Successivamente si realizzi una pagina WEB con una scritta del tipo: "Mancano tot minuti alla mezzanotte" se l'orario è successivo (o uguale) alle ore 12:00, oppure con una scritta del tipo: "La mezzanotte è passata da tot minuti" nel caso contrario (dove tot evidentemente è la quantità di minuti calcolata).

11-02-05-01

ESERCIZI

Creare un progetto Javascript che converta un voto (da 4 a 10) in stringa e che realizzi una pagina WEB che mostri la corrispondenza tra l'input e il valore calcolato.

Possiamo assumere che l'utente inserisca come parte decimale solo 0, 25, 50 o 75.

Possibili output del progetto sono:

"6: sei";
"6,25: sei più";
"6,50: sei e mezzo";
"6,75: sette meno".

11-02-05-05

ESERCIZI

Creare un progetto Javascript che acquisisca nella variabile **nominativo** il nome dell'utente; che – utilizzando un meccanismo che non dia adito a fraintendimenti – riempi una variabile booleana **sessoMaschile** e che stampi quindi un messaggio di benvenuto indirizzato all'utente in accordo con il suo sesso ("Benvenuta Maria!" o "Benvenuto Alessandro!").

11-02-06-01

ESERCIZI

Sviluppare un progetto Javascript che crei un numero casuale superiore a 10000 e che realizzi una pagina WEB che mostri il numero prodotto come una certa quantità di giorni e che successivamente mostri - in un elenco puntato - a quanti anni, quanti mesi e quanti giorni corrisponde quella quantità di giorni.

Si considerino ogni anno di 360 giorni e ogni mese di 30.

11-02-07-01

ESERCIZI

Creare un progetto Javascript che preso un intero n dall'utente, estraiga n numeri casuali compresi tra 1 e 100. Poi si produca una pagina WEB con due elementi di tipo paragrafo. Ciascun elemento presenterà un messaggio per comunicare la percentuale (arrotondata) dei numeri pari e dei numeri dispari estratti.

11-03-01-01

ESERCIZI

Sviluppare un progetto Javascript che preso un valore in input stampi a video il messaggio "Si è sempre a sud di qualcuno" in un popup se il valore era un numero primo e in un elemento di tipo paragrafo altrimenti.

11-03-01-02

ESERCIZI

Creare un progetto Javascript che estragga tanti numeri casuali a due cifre finché la somma degli stessi non superi il valore 1000.

A questo punto verrà mostrato un popup che riporterà la quantità di numeri che si è dovuto generare.

11-03-02-01

ESERCIZI

Creare un progetto Javascript che calcoli la somma dei quadrati dei primi K numeri naturali successivi ad un certo numero naturale N, con N e K presi in input.

11-03-06-01

Creare un progetto Javascript che visualizzi tutti i multipli di N (con N dato dall'utente) che siano compresi tra 50 e 100.

11-03-06-02

ESERCIZI

Creare un progetto Javascript che chieda in input un numero tra 1 e 12 (input controllato) e che produca una pagina WEB che mostri un messaggio (Calibri, verde, 20px, centrato) del tipo: "Il mese di Aprile ha 30 giorni." (in accordo con l'input ricevuto).

Nel caso in cui l'input sia relativo al mese di Febbraio, il progetto chiederà un ulteriore input per l'anno (input controllato) e fornirà un output del tipo: "Il mese di Febbraio (2000) ha 29 giorni."

11-03-06-03

ESERCIZI

Sviluppare un progetto Javascript che simuli il movimento di un cavallo che partendo dalla posizione centrale di una scacchiera 9x9 effettua 10 mosse casuali senza mai uscire dalla scacchiera stessa.

Il progetto creerà una pagina WEB come quella mostrata (testi in Verdana, 20 pixel; distanza dai bordi: 3 pixel; bordi tabella: 2 pixel).

11-03-06-04

1	2	3	4	5	6	7	8	9	10	11
(5,5)	(7,4)	(8,6)	(9,8)	(8,6)	(6,7)	(7,9)	(9,8)	(7,7)	(5,8)	(6,6)

ESERCIZI

Sviluppare un progetto Javascript che realizzi una pagina WEB come quella mostrata.

La pagina mostrerà un menu, le cui voci saranno prese in input dall'utente.

Le voci dovranno essere minimo 2 e massimo 5. Tutte le voci del menu condurranno alla stessa pagina web: www.repubblica.it. I colori del menu saranno marrone e bianco. Ogni testo sarà in grassetto, di dimensione 15 pixel e sarà contenuto in un box 100x35 pixel.

11-03-06-05



ESERCIZI

Sviluppare un progetto Javascript che produca e stampi un numero casuale (tra 1000 e 9999) in numeri romani usando la random una volta sola.

Si faccia riferimento al sistema usato nell'antica Roma (additivo) e non quello usato nel Medioevo (in cui in base alla posizione un simbolo poteva essere aggiunto ma anche sottratto).

11-03-06-06

MMDCLXVIII

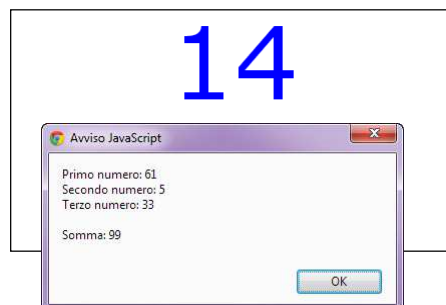
ESERCIZI

Sviluppare un progetto Javascript che produca 3 numeri casuali (compresi tra 1 e 100) la cui somma dia un valore strettamente inferiore a 101.

Il progetto produrrà una pagina web con il numero di tentativi effettuati in blu, Verdana, 100 pixel e un popup come quello presentato in figura.

Svolgere l'esercizio senza usare i vettori.

11-03-06-07



ESERCIZI

Sviluppare un progetto Javascript che produca e stampi una tavola pitagorica come quella in figura. I colori usati sono ciano, blu e azzurro; i testi sono in Verdana (30 pixel) e ciascuna cella è grande 50x70 pixel.

11-03-06-08

1	2	3	4	5	6	7	8	9	10
2	4	6	8	10	12	14	16	18	20
3	6	9	12	15	18	21	24	27	30
4	8	12	16	20	24	28	32	36	40
5	10	15	20	25	30	35	40	45	50
6	12	18	24	30	36	42	48	54	60
7	14	21	28	35	42	49	56	63	70
8	16	24	32	40	48	56	64	72	80
9	18	27	36	45	54	63	72	81	90
10	20	30	40	50	60	70	80	90	100

ESERCIZI

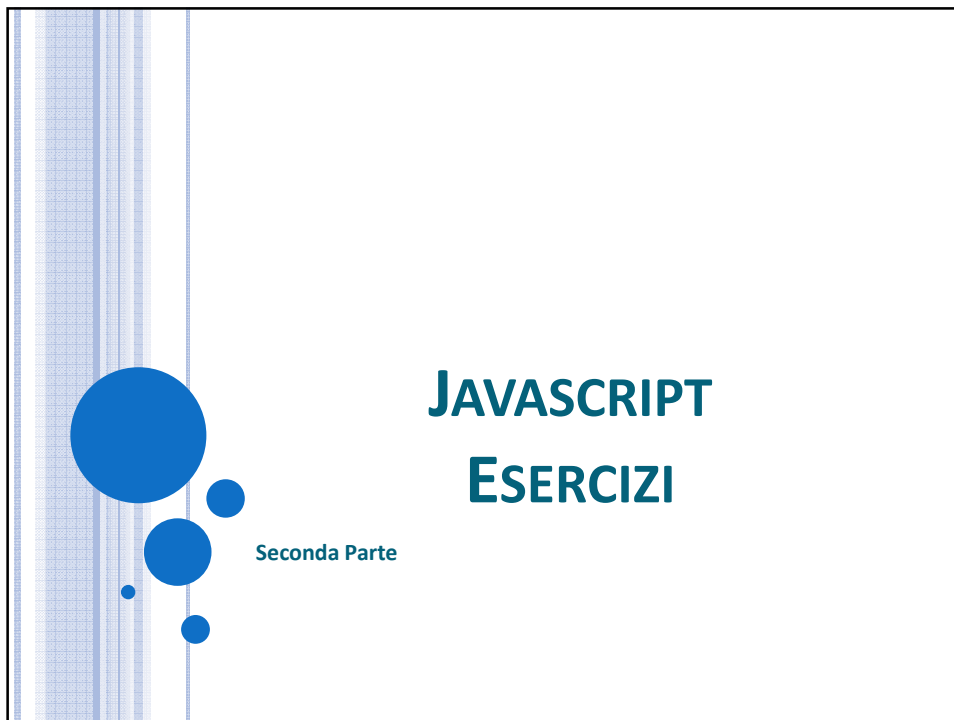
Sviluppare un progetto Javascript che produca e stampi una tabella con un massimo di dieci orari che siano sempre successivi uno all'altro (e non superino la mezzanotte).

Il font usato è Arial ed è grande 50pixel. I colori usati sono rosso e arancione. La tabella dista 25 pixel dal margine superiore della finestra del browser.

11-03-06-10

18:17
22:26
23:25
23:35
23:39
23:41
23:47
23:49
23:53
23:55

1:03
14:47
15:07
20:01
23:45
23:59



JAVASCRIPT → PARTE 2 → I SOTTOPROGRAMMI CC BY-NC-SA

ESERCIZI

Creare una funzione Javascript che determini se un numero è primo o meno e creare poi un progetto Javascript che la usi per presentare a video i primi 10 numeri primi.

11-04-01

Creare una funzione Javascript che restituisca il valore preso in input (come parametro) con massimo due cifre decimali.

Si sviluppi quindi un progetto Javascript che produca 5 numeri casuali con 7 cifre significative, delle quali 5 decimali e che facendo uso della funzione realizzata li stampi nella forma dd,dd.

11-04-02

VERSIONE 1.3 - DIAPOSITIVA 28 ALESSANDRO URSOMANDO

ESERCIZI

Realizzare una funzione Javascript che si chiami **miaRandom**, che prenda due valori x e y e restituisca un intero casuale tra x e y.

Realizzare quindi un progetto Javascript che testi la funzione.

11-04-03

ESERCIZI

Creare una funzione Javascript che preso un numero in input, restituisca la sua prima cifra.

11-04-04

Realizzare un progetto Javascript che acquisito un intero, lo stampi cifra per cifra aumentandone via via la dimensione (si parte da 50 pixel ed ogni volta si aumenta di 20 pixel).

11-04-05

Realizzare una funzione Javascript che avuto un numero intero come parametro restituisca true se il numero è palindromo e false altrimenti.

11-04-06

ESERCIZI

1	2	3	4	5
6	7	8	9	10
11	12	13	14	15
16	17	18	19	20
21	22	23	24	25
26	27	28	29	30
31	32	33	34	35
36	37	38	39	40
41	42	43	44	45
46	47	48	49	50
51	52	53	54	55
56	57	58	59	60
61	62	63	64	65
66	67	68	69	70
71	72	73	74	75
76	77	78	79	80
81	82	83	84	85

Realizzare un progetto Javascript che preso in input un valore, chiamiamolo **n**, trovi due suoi divisori, chiamiamoli **a** e **b**, con **a < b**.

Il sistema dovrà presentare a video una tabella HTML con una quantità di colonne pari ad **a** ed una quantità di righe pari a **b** e mostrare in essa tutti i numeri da 1 a **n**. Nel caso in cui l'utente dia come input un numero primo, il sistema chiederà di ripetere l'inserimento.

11-04-07

ESERCIZI

1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	32
33	34	35	36	37	38	39	40	41	42	43	44	45	46	47	48
49	50	51	52	53	54	55	56	57	58	59	60	61	62	63	64
65	66	67	68	69	70	71	72	73	74	75	76	77	78	79	80
81	82	83	84	85	86	87	88	89	90	91	92	93	94	95	96
97	98	99	100	101	102	103	104	105	106	107	108	109	110	111	112
113	114	115	116	117	118	119	120	121	122	123	124	125	126	127	128
129	130	131	132	133	134	135	136	137	138	139	140	141	142	143	144
145	146	147	148	149	150	151	152	153	154	155	156	157	158	159	160
161	162	163	164	165	166	167	168	169	170	171	172	173	174	175	176
177	178	179	180	181	182	183	184	185	186	187	188	189	190	191	192
193	194	195	196	197	198	199	200	201	202	203	204	205	206	207	208
209	210	211	212	213	214	215	216	217	218	219	220	221	222	223	224
225	226	227	228	229	230	231	232	233	234	235	236	237	238	239	240
241	242	243	244	245	246	247	248	249	250	251	252	253	254	255	256
257	258	259	260	261	262	263	264	265	266	267	268	269	270	271	272
273	274	275	276	277	278	279	280	281	282	283	284	285	286	287	288
289	290	291	292	293	294	295	296	297	298	299	300	301	302	303	304
305	306	307	308	309	310	311	312	313	314	315	316	317	318	319	320
321	322	323	324	325	326	327	328	329	330	331	332	333	334	335	336
337	338	339	340	341	342	343	344	345	346	347	348	349	350	351	352
353	354	355	356	357	358	359	360	361	362	363	364	365	366	367	368
369	370	371	372	373	374	375	376	377	378	379	380	381	382	383	384
385	386	387	388	389	390	391	392	393	394	395	396	397	398	399	400

Realizzare un progetto Javascript che preso in input un valore, chiamiamolo **n**, trovi quella coppia di suoi divisori, chiamiamoli **a** e **b**, con **a < b**, tali che la quantità di divisori minori di **a** e la quantità di divisori maggiori di **b** sia comparabile.

Per esempio, se **n = 400**, i suoi divisori sono: 1, 2, 4, 5, 8, 10, 16, 20, 25, 40, 50, 80, 100, 200, 400.

Due possibili valori utili saranno **a=16** e **b=20** (in quanto i valori minori di **a** sono 6 e i valori maggiori di **b** sono 7).

Il sistema dovrà presentare a video una tabella HTML con una quantità di righe pari ad **a** ed una quantità di colonne pari a **b** e mostrare in essa tutti i numeri da 1 a **n**.

Nel caso in cui l'utente dia come input un numero primo, il sistema chiederà di ripetere l'inserimento.

11-04-08

ESERCIZI

- Medea: 12/8/1990
- Teodolinda: 24/10/1982
- Acheronte: 8/6/1997
- Circe: 17/11/1988
- Morgana: 30/12/1994

Realizzare un progetto Javascript che mostri un elenco puntato come quella in figura.

Il sistema scelga i nomi da un elenco stabilito a design-time proponendoli in modo sempre casuale.

Le date saranno invece scelte a caso in un range di anni indicato dall'utente.

11-04-09

ESERCIZI

Realizzare un progetto Javascript che mostri un popup con una data a caso compresa tra due date prese in input dall'utente.

Si implementi un controllo dell'input che consideri tutti i mesi di 30 giorni e che accolga date degli anni dal 1950 al 2050 (estremi compresi).

11-04-10

ESERCIZI

Creare una funzione Javascript che determini se un anno è bisestile o meno e creare poi un progetto Javascript che la usi per calcolare quanti giorni intercorrono tra due date prese in input.

Si presenti il valore calcolato in Verdana, di colore verde, centrato e di dimensione 100px.

Il progetto può fare uso anche di eventuali altre funzioni che dovessero rendersi necessarie.

11-04-11

ESERCIZI

Creare un progetto Javascript che acquisisca una quantità imprecisata di numeri interi positivi (-1 per uscire) e dopo ogni input stampi a video:

- la quantità di numeri gestiti
- la loro somma
- la loro media
- il maggiore
- il minore.

Una volta acquisito l'ultimo numero (-1) si esegua la somma di tutti e 5 gli output prodotti e si verifichi se tale valore sia magico o meno (stampando a video true o false).

Si definiscono magici tutti i numeri con esattamente 3 cifre decimali e la cui somma (di tutte le cifre: decimali e non) dia un numero dispari.

Si adoperino le funzioni.

11-04-12

ESERCIZI

Sviluppare una funzione Javascript che preso un numero in input (sia intero che decimale) lo restituisca senza la virgola.
Per esempio: 21.89 restituisce 2189.

11-04-13

ESERCIZI

Creare una funzione Javascript che avuta una stringa come parametro restituisca vero se e solo se i caratteri alfabetici inseriti sono tutti maiuscoli.

Realizzare quindi un progetto Javascript che – facendo uso di tale funzione – continui a chiedere un input qualsiasi all'utente fino all'inserimento di una stringa tutta in maiuscolo.

11-45-010

Creare una funzione Javascript che restituisca la stringa ricevuta input (come parametro) epurata di tutti gli eventuali caratteri numerici presenti.

Si realizzi quindi un progetto Javascript che chiesto ed ottenuto un input alfanumerico, presenti il risultato prodotto dalla funzione sviluppata.

11-45-020

Creare una funzione Javascript che produca e restituisca un codice casuale della forma XYYY (dove XX sono due vocali e YYY un numero a 3 cifre).

Realizzare quindi un progetto Javascript che – facendo uso di tale funzione – presenti il codice di un articolo all'utente e che chieda in input la quantità ed il prezzo unitario fino a raggiungere la cifra di 1000 euro.

11-45-030

ESERCIZI

Sviluppare una funzione Javascript che restituisca vero se la stringa passata come parametro è palindroma, falso altrimenti.

Si realizzi un progetto Javascript che usi tale funzione con una stringa presa in input dall'utente.

11-45-040

ESERCIZI

Sviluppare un progetto Javascript che presa una parola qualsiasi la stampi con uno dei set di immagini fornite.

La scelta del set di immagini deve essere casuale.

Il controllo dell'input ammetterà solo l'inserimento di lettere (maiuscole o minuscole).

11-45-050

A L E S S A N D R O

U R S O M A N D O

ESERCIZI

Realizzare un progetto Javascript che presenti una griglia come quella mostrata.
Si raccomanda l'utilizzo dell'elemento `div` (non usare l'elemento `table`).

11-45-060



ESERCIZI

Risolvere l'esercizio precedente facendo in modo che l'immagine della cella centrale sia scelta a caso dal sistema.

Non ci sono vincoli su dove dovranno comparire tutte le altre immagini.

SUGGERIMENTO: poiché tutte le immagini devono comparire una ed una sola volta nella griglia è consigliabile mantenere un ordine.

11-45-070



ESERCIZIO

Creare una funzione Javascript che calcoli e restituisca la quantità di giorni che intercorrono tra due date prese in input come parametri.

Sviluppare quindi un progetto Javascript che produca 2 date casuali e - facendo uso di tale funzione - mostri in output le due date e la quantità di giorni che intercorre tra le stesse.

Supponiamo che le date siano comprese tra il 1° Gennaio 2000 ed il 31 dicembre 2010.

Il progetto può fare uso anche di eventuali altre funzioni che dovessero rendersi necessarie.

11-05-01

ESERCIZIO

Creare una funzione Javascript che presa in input una data (come parametro) verifichi che sia una data relativa ad un giorno feriale.

Creare quindi una funzione Javascript che presa in input una data (come parametro) la modifichi fino a farla diventare una data relativa ad un giorno feriale (qualora non lo sia già). Si usi la funzione precedentemente sviluppata.

Sviluppare quindi un progetto Javascript che chieda all'utente una data di prenotazione e che stampi in output i seguenti messaggi.

“A partire dalla data inserita ci occorrono 10 giorni lavorativi.”

“La consegna è prevista per il giorno **gg/mm/aaaa**”.

Dove la data **gg/mm/aaaa** sia quella calcolata.

11-05-02

ESERCIZI

Realizzare un progetto Javascript che dichiari un vettore di 10 numeri reali a piacere. Si mostrino poi a video 5 componenti di questo vettore a caso.

11-06-01

Sviluppare un progetto Javascript che prenda in input due numeri positivi (uno successivo all'altro) e che crei un vettore di 10 numeri interi compresi nell'intervallo avuto in input.

Successivamente si mostrino tutti gli elementi del vettore in ordine decrescente.

11-06-02

ESERCIZI

Bari	26	32	35	35	36	46	55	63	84	87
Cagliari	13	17	20	36	39	44	57	67	68	79
Firenze	8	36	39	45	58	61	63	75	77	83
Genova	4	7	28	40	57	67	71	78	80	87
Milano	3	8	51	68	73	78	78	82	86	88
Napoli	1	2	10	14	22	39	55	67	84	87
Palermo	24	35	43	47	61	62	77	79	86	90
Roma	4	13	16	21	30	33	42	45	61	84
Torino	14	14	27	35	41	45	54	72	81	82
Venezia	11	21	46	53	57	64	68	77	82	89

Realizzare un progetto Javascript che facendo uso dei vettori produca uno schema di estrazioni del lotto.

11-06-03

ESERCIZI

Valori tra 43 e 96

62	64	47	90	54	64	46	50	88	80
----	----	----	----	----	----	----	----	----	----

Sviluppare una funzione Javascript che riempia un vettore (preso come parametro) con 10 numeri naturali compresi in un intervallo (anch'esso preso come parametro).

La funzione restituirà **false** nel caso in cui i parametri non dovessero consentire il lavoro (vettore di dimensione diversa da 10 o intervallo non corretto).

Sviluppare quindi un progetto Javascript che usi tale funzione e che mostri il contenuto del vettore; il range da dare in input alla funzione sarà stabilito da due valori casuali a due cifre.

11-06-04

ESERCIZI

Valori tra 80 e 99

84	85	88	90	92	93	95	97	98	99
----	----	----	----	----	----	----	----	----	----

Risolvere l'esercizio precedente facendo in modo che la funzione produca un vettore senza duplicati (in questo caso bisognerà gestire anche la cardinalità dell'intervallo).

11-06-05

ESERCIZI

Creare una funzione Javascript che preso un vettore come parametro (non necessariamente vuoto), lo restituisca con cento elementi numerici a due cifre.

Creare una seconda funzione Javascript che preso un vettore (di qualsiasi dimensione) ed un valore qualsiasi restituisca il numero di occorrenze di quel valore nel vettore dato.

Sviluppare quindi un progetto Javascript che utilizzi le due funzioni facendo scegliere all'utente il valore da cercare nel vettore .

Si verifichi la correttezza dell'input.

Si produca in output il valore calcolato e una tabella che permetta di riscontrarne la correttezza.

11-06-06

Occorrenze di 55 nel vettore: 2

39	26	93	25	36	22	16	49	38	37	86	47	89	55	74	70	60	61	91	33
18	97	60	27	23	43	14	78	90	10	35	14	74	21	16	38	92	47	82	93
61	88	34	68	68	33	97	52	14	29	80	10	89	82	49	15	83	91	99	34
15	90	79	45	49	17	69	57	96	51	45	86	27	33	78	77	55	44	79	78
95	20	73	53	22	36	54	48	48	89	56	30	22	10	40	42	42	78	12	44

ESERCIZI

582|712|503|494|678|487|923|246|476|694

582|712|503|494|678|487|246|476|694

Sviluppare una funzione in Javascript che preso un vettore come parametro, lo restituisca senza il suo valore maggiore.

Sviluppare quindi un progetto Javascript che crei un vettore con 10, 20 o 30 (scelta casuale) numeri casuali a tre cifre, lo stampi così com'è e poi lo ristampi dopo aver usato su di esso la funzione di cui sopra.

11-06-07

405|335|730|585|875|130|204|240|788|526

405|335|730|585|875|204|240|788|526

Svolgere nuovamente l'esercizio precedente con le seguenti modifiche.

Il valore da eliminare è il minore (invece del maggiore).

La funzione di stampa del vettore avrà un secondo argomento opzionale, il quale indicherà la cella del vettore da presentare evidenziata (nel nostro caso quella con il valore minore).

11-06-08

ESERCIZI

Svilappare un progetto Javascript che presa una prima data in input e creata una seconda data casuale dello stesso anno e successiva alla prima, crei un vettore con tutte le date nell'intervallo definito (estremi compresi) e poi stampi un calendario con le date del vettore come quello in figura.

11-06-09

Venerdi	15	Ottobre	2010
Sabato	16	Ottobre	2010
Domenica	17	Ottobre	2010
Lunedì	18	Ottobre	2010
Martedì	19	Ottobre	2010
Mercoledì	20	Ottobre	2010
Giovedì	21	Ottobre	2010
Venerdi	22	Ottobre	2010
Sabato	23	Ottobre	2010
Domenica	24	Ottobre	2010
Lunedì	25	Ottobre	2010
Martedì	26	Ottobre	2010

ESERCIZI

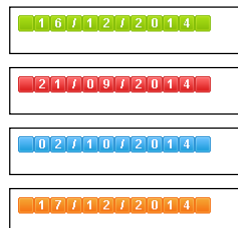
(a)
Svilappare una funzione Javascript che produca una data casuale (dal 1/1/2013 al 31/12/2014).

(b)
Usando tale funzione, sviluppare un progetto Javascript che crei un vettore di 10 date e

(c)
cerchi in questo vettore la data maggiore.

(d)
Tale data sarà messa a video con un set di immagini a caso tra quelli forniti.

11-06-10



(a): 5 punti

(b): 1 punto

(c): 3 punti

(d): 1 punto

PUNTO (1)

Realizzare una classe **Punto** in Javascript per la definizione di oggetti di tipo punto sul piano cartesiano. La classe avrà gli attributi ascissa e ordinata ed i seguenti metodi: setAscissa e setOrdinata, getAscissa e getOrdinata, setPunto (che prenderà due argomenti) e un metodo getFormatoStringa. Questo ultimo metodo restituirà una stringa del tipo (5,5) dove 5 è il valore dell'ascissa e dell'ordinata. Si realizzi inoltre una piccola porzione di codice che testi tutte le funzionalità della classe.

11-07-01

PUNTO (2)

Realizzare una classe **Punto** in Javascript per la definizione di oggetti di tipo punto sul piano cartesiano. La classe avrà gli attributi ascissa e ordinata ed i metodi: setAscissa, setOrdinata, getAscissa, getOrdinata, setPunto e getFormatoStringa. Si realizzi la classe in modo tale che siano disponibili tre costruttori: uno senza argomenti, uno con due argomenti di tipo numerico e uno con un solo argomento di tipo Punto. Si realizzi il metodo setPunto in modo tale da essere invocato o con una coppia di numerici o con un solo argomento di tipo Punto. Si realizzi inoltre una piccola porzione di codice che testi tutte le funzionalità della classe.

11-08-01

La condizione qui di seguito è vera se e solo se la variabile **x** è di tipo **Punto**.

```
(x instanceof Punto)
```

CONTATORE

Realizzare una classe **Contatore** in Javascript.

La classe gestirà un contatore che potrà essere inizializzato a un certo valore o a zero (valore di default); impostato ad un certo valore; letto con un apposito metodo; confrontato con altro valore (minore di, maggiore di, uguale); incrementato di uno (valore di default) o di una certa quantità indicata, stampato nel formato di seguito riportato: [18].

Si realizzi inoltre una piccola porzione di codice che testi tutte le funzionalità della classe.

11-08-02

SEMAFORO

Realizzare una classe **Semaforo** in Javascript con i colori rosso, giallo e verde che tenga il rosso ed il verde per dieci secondi ed il giallo per tre secondi.

La classe avrà gli attributi stato e secondiTrascorsi ed i metodi getStato, avanzaStato e avanzaDiUnSecondo.

Quest'ultimo metodo, qualora sia giunto il momento di cambiare colore, procederà invocando l'opportuno metodo.

Si realizzi inoltre una piccola porzione di codice che istanzi un oggetto di tipo Semaforo e ne stampi subito il valore. Successivamente (per tre volte) si simuli il trascorrere di 6 secondi e si stampi nuovamente il valore del semaforo.

11-08-03

PILA (1)

Realizzare una classe **Pila** (struttura LIFO) in Javascript.

La classe espone i metodi **push** (deposita un valore in testa alla pila),
pop (toglie un valore dalla testa della pila),
isEmpty (restituisce true se la pila è vuota)
e **toString**

(restituisce una stringa che elenca i valori della pila dal primo all'ultimo).

È consentito fare uso di un array non dimensionato, della proprietà `length` della classe `Array`, delle operazioni di `push` e `pop` valide sui vettori Javascript.

Si realizzi inoltre una piccola porzione di codice che testi tutta la classe.

11-08-04

PILA (2)

Realizzare una classe **Pila** (struttura LIFO) in Javascript.

La classe espone i metodi **push** (deposita un valore in testa alla pila),
pop (toglie un valore dalla testa della pila),
isEmpty (restituisce true se la pila è vuota)
isFull (restituisce true se la pila è piena)
e **toString**

(restituisce una stringa che elenca i valori della pila dal primo all'ultimo).

Si farà uso di un array a dimensione fissa e si farà a meno di adoperare la proprietà `length` della classe `Array` e le operazioni di `push` e `pop` valide sui vettori Javascript.

Si realizzi inoltre una piccola porzione di codice che testi tutta la classe.

11-08-05

CODA (1)

Realizzare una classe **Coda** (struttura FIFO) in Javascript.

La classe espone i metodi **put** (deposita un valore alla fine della coda),
get (toglie un valore dalla testa della coda),
isEmpty (restituisce true se la coda è vuota)
e **toString**

(restituisce una stringa che elenca i valori della coda dal primo all'ultimo).

È consentito fare uso di un array non dimensionato, della proprietà `length` della classe `Array`, delle operazioni di `push` e `shift` valide sui vettori Javascript.

Si realizzi inoltre una piccola porzione di codice che testi tutta la classe.

11-08-06

CODA (2)

Realizzare una classe **Coda** circolare (struttura FIFO) in Javascript.

La classe espone i metodi **put** (deposita un valore alla fine della coda),
get (toglie un valore dalla testa della coda),
isEmpty (restituisce true se la coda è vuota)
isFull (restituisce true se la coda è piena)
e **toString**

(restituisce una stringa che elenca i valori della coda dal primo all'ultimo).

Si farà uso di un array a dimensione fissa e si farà a meno di adoperare la proprietà `length` della classe `Array` e le operazioni di `push` e `shift` valide sui vettori Javascript.

Si realizzi inoltre una piccola porzione di codice che testi tutta la classe.

11-08-07

MYORARIO

Realizzare una classe **myOrario** in Javascript che abbia gli attributi **ora** e **min** (non è necessario gestire i secondi) e che esponga metodi per impostare e leggere ciascun attributo più i metodi descritti di seguito.

Si richiede un metodo che restituisca una stringa che rappresenti l'orario (per esempio 15:30, oppure 19:04).

Si richiede un metodo che imposti l'orario sulla base degli argomenti passati: saranno validi
un parametro di tipo **myOrario**
oppure due parametri di tipo **number**.

Si richiede la realizzazione di un costruttore che lavori esattamente come il metodo appena descritto.

Si realizzi inoltre una piccola porzione di codice che testi tutta la classe.

11-08-08

ESERCIZI

Persona
nome
cognome
dataDiNascita
costruttore()
costruttore(n,c,d)
costruttore(p)
setNome(x)
setCognome(x)
setDataDiNascita(x)
getNome()
getCognome()
getDataDiNascita()
getEta()
toString()

Realizzare una classe **Persona** come quella descritta dallo schema e successivamente estendere da essa una classe **Impiegato** che aggiunga
gli attributi mansione e stipendio,
i metodi necessari a gestire
l'incapsulamento di questi nuovi attributi
e che implementi nuove versioni del costruttore e del metodo **toString()**.

11-09-01

ESERCIZI

Estendere la classe **Impiegato** dell'esercizio precedente realizzando la classe **Manager**.
Un manager può avere una segretaria.

11-09-02

ESERCIZI

Realizzare una classe denominata **Classe** la quale contenga gli attributi **annoDiCorso** e **sezione**.
Realizzare inoltre una classe **Studente** che estenda la classe **Persona** (già realizzata negli esercizi precedenti).
Questa classe aggiunge a quanto contenuto in **Persona** un attributo (classe) di tipo **Classe**.

11-09-03

ESERCIZI

Pulsante
etichetta
stato
costruttore()
costruttore(e)
costruttore(p)
setEtichetta(x)
setStatoGiu()
setStatoSu()
getEtichetta()
isStatoGiu()
isStatoSu()
toString()

Realizzare una classe **Pulsante** come quella descritta dallo schema.

Il valore dell'attributo stato può essere "su" o "giu" e sarà gestito mediante appositi metodi.

Il formato stringa di un oggetto Pulsante sarà simile a **PUSH []** oppure a **PUSH []**.
(codice ASCII: 238).

Successivamente estendere da essa una classe **PulsanteAvanzato** il cui attributo stato potrà assumere i valori "su", "giu" e "metà".

11-09-05

ESERCIZI

Realizzare una classe **Ingrediente** con gli attributi nome e quantità.

Realizzare quindi una classe **Ricetta** con gli attributi nome, procedimento e listaIngredienti (quest'ultimo attributo è un vettore di elementi di tipo Ingrediente).

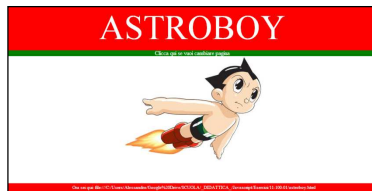
Estendere infine questa classe per realizzare la classe **VideoRicetta** che aggiunge gli attributi durata e pathFile.

11-09-05

ESERCIZI

Realizzare un progetto Javascript che presenti le pagine HTML relative ai cartoni animati forniti. Ciascuna pagina sarà composta da un titolo (per il nome del cartone animato), un footer (per indicare l'url del file presentato), una immagine (raffigurante il cartone animato) e una banda con del testo che - se cliccato - presenti un'altra pagina a caso del progetto (diversa da quella attuale). Il progetto deve avere la pagina index.html e una pagina per ciascun cartone.

11-100-01



ESERCIZI

Realizzare completamente in Javascript il progetto mostrato in figura. Devono essere visualizzate le sei immagini fornite nelle dimensioni (130x100) e deve essere gestito il focus ottenuto dalle stesse come di seguito indicato.

Ogni volta che un'immagine riceve il focus, il titolo della pagina WEB riporta il nome del soggetto dell'immagine (la squadra, Lupin, Jigen, Goemon, Fujico, Zenigata).

Quando nessuna immagine ha il focus il titolo della pagina web diventa: "Esercizio".

11-100-02



ESERCIZIO

Clicca qui per entrare

Clicca qui per entrare

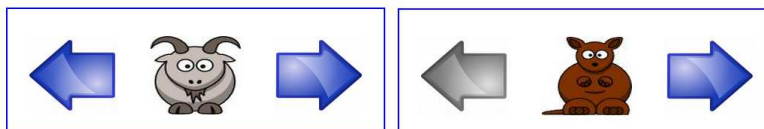
Sviluppare un progetto Javascript che presenti l'immagine di un pulsante il quale, in accordo con il movimento del mouse, si presenti nelle sue due versioni: premuto o da premere.

Quando l'utente clicca sul pulsante il sistema chiede in input il nome dell'utente e lo confronta con gli elementi di un vettore di nomi inizializzato con i nomi di Matteo, Giacomo, Giovanni e Luca.

Se il nome inserito è nell'elenco si presenta un popup di accesso avvenuto altrimenti si presenta un popup di accesso rifiutato.

11-100-03

ESERCIZIO



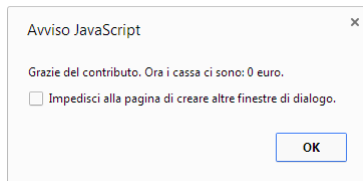
Sviluppare un progetto Javascript che presenti l'immagine di un animale tra due frecce.

Cliccando sulle frecce l'utente si sposta tra le immagini disponibili. Quando non ci sono più animali a sinistra (a destra) di quello mostrato la freccia a sinistra (a destra) diventa grigia.

Quando l'utente clicca su una freccia grigia, riceve un messaggio popup del tipo: "prima immagine" o "ultima immagine".

11-100-04

ESERCIZIO



Sviluppare un progetto Javascript che gestisca una cassa.

Cliccando sul simbolo della moneta si mettono in cassa due euro e cliccando sul cuore si tolgono dalla cassa due euro.

Ogni volta che si modifica la cassa appare un messaggio come quello in figura. Quando si clicca sul cuore con la cassa vuota compare un messaggio di avviso.

11-100-05

ESERCIZIO



Sviluppare un progetto Javascript che gestisca una cassa.

Cliccando sul simbolo della moneta si mettono in cassa due euro e cliccando sul cuore si tolgono dalla cassa due euro.

Ogni volta che si modifica la cassa le immagini in basso vengono modificate affinché rappresentino la quantità di denaro in cassa.

Quando si clicca sul cuore con la cassa vuota compare un messaggio di avviso.

11-100-06

ESERCIZIO

Medici senza frontiere
Emergency
Save the children
Amnesty International



Sviluppare un progetto Javascript che presenti un menu come quello in figura.

Cliccando su una voce questa diventa rossa e cliccando sul pulsante si apre (in un'altra scheda) il sito relativo alla voce selezionata in rosso.

11-100-07