

Il proprietario di un'azienda vi incarica di realizzare un sistema software per calcolare il salario settimanale dei suoi dipendenti. In parole povere, il sistema deve rispondere a domande del tipo:

“La settimana scorsa, il dipendente Marco Rossi ha lavorato 8 ore al giorno da lunedì a giovedì, mentre venerdì ha lavorato solo 4 ore. Sabato ha lavorato 8 ore più 2 ore di straordinario. La domenica non ha lavorato perché è il suo giorno di riposo. Quanti soldi gli spettano?”

Vi vengono fornite le seguenti indicazioni sui dipendenti e sul calcolo dei salari settimanali:

Ordine di svolgimento consigliato: VERDE, GIALLO, ARANCIONE, ROSSO.

[CST] Campi e costruttori	[EXC] Controlli ed eccezioni	[STC] Campi e metodi statici
[OBJ] Override di Object	[CLA] Classi ed ereditarietà	[ALG] Algoritmi e calcoli
[ENM] Enumerazioni	[COL] Collezioni e mappe	[REC] Ricorsione
[---] Requisito soddisfatto dal codice di partenza o per conseguenza di altri requisiti.		

Dati dei dipendenti

- [CST] Il proprietario deve poter inserire nel sistema il nome e il cognome dei suoi dipendenti.
- [CST|EXC] Il nome e il cognome devono essere obbligatori: non possono essere nulli o vuoti.
- [---] Il sistema assegnerà automaticamente un ID numerico a ogni dipendente inserito.
- [CST|STC] L'ID deve essere univoco e crescente: il primo dipendente ha ID 1, il secondo ha ID 2, ecc.
- [OBJ] Possono esserci dipendenti omonimi: i dipendenti sono identificati solo dal loro ID numerico.
- [OBJ] Dev'essere possibile convertire in un'unica riga di testo i dati di un dipendente, incluso l'ID.

Paga oraria

- [CLA] Ci sono due tipi di dipendenti: impiegati e dirigenti.
- [STC] Attualmente, la paga oraria standard di tutti gli impiegati è di €12.
- [STC] Il proprietario dovrà essere libero di modificare tale paga oraria standard a suo piacimento.
- [STC|EXC] Per legge, la paga oraria standard degli impiegati non può mai scendere al di sotto di €10.
- [STC] La paga oraria standard dei dirigenti è sempre pari al 150% di quella degli impiegati.

Giorni e ore di lavoro

- [---] I dipendenti possono lavorare in qualunque giorno della settimana.
- [---] In una certa settimana, i diversi dipendenti non lavorano necessariamente negli stessi giorni.
- [---] In un certo giorno, i diversi dipendenti non lavorano necessariamente per le stesse ore.
- [ALG|ENM] Da lunedì a venerdì, le prime 8 ore di lavoro sono retribuite secondo la paga oraria standard.
- [ALG] Se un dipendente lavora meno di 8 ore in un giorno, le ore mancanti non vengono retribuite.
- [---] I giorni della settimana in cui un dipendente non lavora non sono retribuiti.
- [ALG] Le eventuali ore di straordinario (oltre le 8 ore di lavoro standard) vengono pagate il 10% in più.
- [EXC] In un giorno di lavoro, ciascun dipendente può lavorare al massimo 10 ore.
- [EXC|COL] In una settimana di lavoro, ciascun dipendente può lavorare al massimo 6 giorni.
- [ALG|ENM] Il sabato, tutte le ore di lavoro vengono pagate un ulteriore 15% in più.
- [ALG|ENM] La domenica, tutte le ore di lavoro vengono pagate un ulteriore 20% in più.

Team di lavoro

- [COL] Ogni dirigente può essere a capo di un team di suoi diretti sottoposti.
- [COL|CLA] Al team di un dirigente possono essere aggiunti sia impiegati che altri dirigenti.
- [---] I dirigenti che fanno parte di un team possono a loro volta essere a capo di un loro team.
- [COL|EXC] Al team di un dirigente possono essere aggiunti al massimo 4 membri.
- [COL|EXC] Un dirigente non può essere un sottoposto diretto di sé stesso.
- [COL|EXC] I dipendenti non possono essere aggiunti più di una volta allo stesso team.
- [---] I dipendenti possono liberamente essere aggiunti a diversi team.
- [ALG|COL] Ogni settimana, un dirigente riceve un bonus di €25 per ogni membro del suo team.
- [ALG|CLA|REC] Se un membro è un dirigente, il bonus aumenta del 20% del bonus di quest'ultimo.
- [COL|CLA|REC|EXC] Un dirigente non può essere aggiunto al team di un suo sottoposto diretto o indiretto.

Indicazioni tecniche per lo svolgimento

L'esercizio è già parzialmente svolto: all'interno del progetto trovate alcune classi ed enumerazioni già pronte. La richiesta che vi viene fatta è la seguente:

- Sviluppate una o più sottoclassi (dirette o indirette) della classe **Worker** che permettano di soddisfare il più fedelmente possibile i requisiti del cliente.

Non è obbligatorio soddisfare tutti i requisiti, e nemmeno soddisfarli nell'ordine in cui li ho indicati: cercate di capire quali sono quelli alla vostra portata e fate semplicemente ciò che potete. In ogni caso, siete tenuti a completare l'esercizio rispettando i seguenti limiti:

- Nella classe **Worker** potete solo scrivere costruttori e ridefinire metodi della classe **Object**.
- A parte ciò, non vi è permesso modificare la classe **Worker** in nessun altro modo.
- Per il calcolo del salario dovete implementare il metodo astratto **computeSalaryForWeek**.
- Non vi è permesso modificare l'enumerazione **WeekDay** in alcun modo.
- Nella classe **WorkWeek** potete solo aggiungere controlli ed eccezioni ai metodi presenti.

Per il resto, siete liberi di usare qualsiasi strumento a vostra disposizione. Potete utilizzare tutte le classi, interfacce, eccezioni, ecc. della **libreria standard** di Java, e se lo ritenete opportuno (ma non è strettamente necessario) potete anche sviluppare classi, interfacce, eccezioni, ecc. aggiuntive.

Non è richiesto di scrivere un metodo **main** che esemplifichi l'uso delle vostre classi e dei vostri metodi, ma è caldamente consigliato farlo per testare il comportamento del vostro codice.

Non occorre usare la classe **BigDecimal** per i calcoli economici: usate pure il tipo **float** (o **double**).

Consigli

I requisiti dell'esercizio sembrano tanti, ma non spaventatevi: molti di essi sono più semplici di quello che sembrano; solo alcuni sono più impegnativi da soddisfare. Prima di fare qualsiasi cosa, leggeteli con calma e cercate di comprenderli al meglio. Non sottovalutate questo aspetto: se non capite cosa vi viene chiesto di fare, rischiate di svolgere lavoro inutile che non soddisfa le richieste dei vostri committenti (cosa che in genere non vi attira le loro simpatie).

Prima di cominciare a sviluppare, analizzate attentamente i codici già presenti e cercate di capire in che modo potreste usarli per proseguire. Saper leggere, utilizzare, e modificare codici di altri è spesso più importante che saper scrivere nuovi codici da zero: se lavorerete per aziende non troppo giovani, vi capiterà spesso di dover lavorare su progetti già avviati da altri sviluppatori.

Non dimenticate di consultare la documentazione ufficiale di Java: potreste scoprire classi e metodi già pronti che vi permettono di svolgere un certo compito in poche righe di codice, oppure scoprire che state causando malfunzionamenti usando una classe o un metodo in modo inappropriato.

Ricordatevi dell'esistenza del debugger: in caso di bug, analizzate il vostro codice riga per riga.

Siete liberi di consultarvi tra voi, chiedermi aiuto, e copiare codice da internet o da altre fonti: sul lavoro dovrete spesso chiedere aiuto ai vostri colleghi e superiori, oppure cercare spunti e soluzioni su internet o altrove. Cercate comunque di trovare soluzioni con le vostre forze e i vostri mezzi: oggi voi potete contare sugli altri, ma domani potrebbero essere gli altri a dover contare su di voi. Qualunque cosa facciate, dovete sempre essere in grado di spiegare cosa avete scritto e perché.