

## CORSO DI FORMAZIONE

### “Programmatore Informatico in Java”

(QUALIFICAZIONE PROGRAMMATORE INFORMATICO - PI K1.12)

*Corso di Formazione autorizzato dalla Regione Lazio con Esame di Qualificazione Professionale*

#### DESCRIZIONE

Il Programmatore Informatico è una figura professionale ampiamente richiesta dal mondo delle aziende nell’ambito dei servizi digitali, in grado di realizzare sulla base di specifiche date, applicazioni software, app e siti web utilizzando uno o più linguaggi di programmazione (*coding* in Java e applicazioni web *full stack* con Spring Boot e Angular).

Il Programmatore Informatico qualificato possiede competenze professionali specifiche nella progettazione e sviluppo del software e delle interfacce, nell’interrogazione e gestione dei dati (*database management system*), nelle tecniche di *cybersecurity* e nella verifica delle funzionalità del software, nell’uso di tecniche di testing fino al deployment del software sviluppato.

#### DESTINATARI

Giovani non qualificati e disoccupati di lunga durata con competenze informatiche di base con necessità di qualificarsi/riqualificarsi sui linguaggi di programmazione richiesti dalle aziende.

#### REQUISITI OBBLIGATORI

- Titoli di Studio:
  - Diploma di scuola secondaria di secondo grado
- In caso di titolo di studio conseguito all'estero, è necessario presentare una dichiarazione di valore o un documento equipollente, che ne attesti la corrispondenza di valore con i titoli rilasciati nello Stato di provenienza, ai fini della verifica dei livelli di scolarizzazione.
- Per i cittadini stranieri, conoscenza della lingua italiana almeno al livello B1 del Quadro Comune Europeo di Riferimento per le Lingue, ferma restando l’obbligatorietà delle prove valutative in sede di selezione, qualora il candidato non disponga già di attestazione di valore equivalente.
- I cittadini extracomunitari devono disporre di regolare permesso di soggiorno, valido per l’intera durata del percorso.

#### STRUTTURA DEL CORSO

Durata ore:

- 430 ore Corso “Programmatore Informatico in Java”, parte in presenza e parte in e-learning sincrono
- 140 ore di Tirocinio Curriculare
- 30 ore Modulo Integrativo di Coaching & Job Placement, di cui:
  - 8 ore Orientamento Specialistico
  - 22 ore Accompagnamento al Lavoro

Durata complessiva: 600 ore / 8 mesi

Le lezioni in aula si svolgono con una frequenza di 3 gg a settimana tra il lunedì e il venerdì, e hanno durata da 4 o 6 ore.

Un numero limitato di ore di didattica potrà svolgersi in modalità Formazione a Distanza (FAD) in e-learning sincrono. Il Calendario Didattico è suscettibile di variazioni. Eventuali cambiamenti verranno comunicati agli Allievi e alla Regione Lazio.

#### ESAME DI QUALIFICAZIONE PROFESSIONALE

Gli allievi sono ammessi all’esame finale a condizione di avere frequentato almeno l’80% delle ore complessive del percorso formativo.

Le prove finali si svolgono di fronte ad una Commissione Esaminatrice composta da un rappresentante della Regione Lazio, che la presiede; due rappresentanti dei docenti; il Responsabile Didattico; un rappresentante del Ministero dell'Istruzione e un rappresentante del Ministero del Lavoro, oltre ai rappresentanti dalle organizzazioni degli imprenditori e dei lavoratori.

LIVELLO EQF DELLA QUALIFICAZIONE: 5

Codice Profilo: K1.12 - Programmatore Informatico

Diploma finale in esito ad esame ai sensi del D.lgs. 13/2013

#### MODALITÀ DIDATTICHE

Lezioni frontali, esercitazioni guidate individuali e di gruppo, simulazioni, analisi di casi e problem solving. Le aule sono dotate di:

- Postazioni docenti con leggìo interattivo, computer MSI All-in-one Adora24G 2NC e sgabello;
- Computer per gli studenti MSI All-in-one Adora20G;
- LIM (Lavagna Interattiva Multimediale), munita di pennarello elettronico e collegate a videoproiettori Epson ultracorto EB 585 Wi.

Modalità di valutazione degli apprendimenti

Test a risposte multiple e discussione in aula al termine di ogni unità didattica, realizzazione di un Project Work e simulazione di casi.

#### SEDE DEI CORSI

Viale Filippo Tommaso Marinetti, 221 - 00143 Roma

Tel: 06 39746618 | Fax: 06 97749271 - [www.accademiainformatica.com/corsi/](http://www.accademiainformatica.com/corsi/)

E-mail: [info@accademiainformatica.com](mailto:info@accademiainformatica.com)

#### DOCENTI DEL CORSO

Andrea Dimitri, *Esperto ICT e Sicurezza Informatica. Esperto in big data analysis e machine learning. Esperto full stack developer. Docente dell'Università degli Studi di Roma "Tor Vergata"*

Franco Arcieri, *Ingegnere elettronico esperto in sicurezza informatica e di rete. Professore di Sistemi Cooperativi Distribuiti presso la Facoltà di Scienze dell'Università degli Studi di Roma "Tor Vergata".*

Simone Ferretti, *Consulente e Formatore in Sicurezza sul Lavoro (d. lgs 81/08)*

Laura Spila, *Orientatore Professionale e Operatore del Mercato del Lavoro Specialistico*

#### SELEZIONE E AMMISSIONE

L'ammissione al Corso è subordinata ad una positiva valutazione del titolo di studio richiesto come Requisito di ammissione e del curriculum del candidato nonché al successivo superamento di un test informatico di base e un colloquio motivazionale/attitudinale.

La Direzione del Corso nominerà un'apposita Commissione, incaricata di valutare preventivamente i titoli presentati dai candidati e di svolgere le prove. Al termine di ciascuna selezione i candidati riceveranno, tramite l'utilizzo dell'indirizzo di posta elettronica fornito, nota della loro ammissione o esclusione ad insindacabile giudizio della Commissione.

#### COSTO DEL CORSO

€ 4.100,00

10% di sconto in caso di pagamento in un'unica soluzione



PROGRAMMA IN SINTESI		
MODULO	DURATA	CONTENUTI FORMATIVI
MODULO INTRODUTTIVO Inquadramento della Professione Inquadramento dei sistemi informatici e basi di cybersecurity	6 ore	Inquadramento della professione
	22 ore	Inquadramento dei sistemi informatici
	15 ore	Basi di cybersecurity
MODULO 1. Scelta delle modalità di sviluppo del software	37 ore	Definire le modalità di sviluppo del Software
MODULO 2. Sviluppo delle componenti software e delle interfacce	112 ore	Svolgere attività di coding - Linguaggi a oggetti e procedurali
	15 ore	Svolgere attività di coding - Linguaggio di scripting e markup
	22 ore	Applicare principi di business logic e responsive design
MODULO 3. Modellazione, interrogazione e gestione dei dati	22 ore	Applicare principi di universal design
	112 ore	Modellizzare, interrogare e gestire i dati
MODULO 4. Testing e Deployment del software sviluppato	22 ore	Testare il prodotto sviluppato
	37 ore	Definire e condurre il Deployment
MODULO SICUREZZA Sicurezza sul Lavoro	8 ore	Operare in sicurezza nel luogo di lavoro
TIROCINIO IN AZIENDA	140 ore	Attività Formativa in Azienda
<b>Totale corso</b>	<b>570 ore</b>	

MODULO	DURATA	CONTENUTI DI DETTAGLIO
MODULO INTRODUTTIVO Inquadramento della Professione Inquadramento dei sistemi informatici e basi di cybersecurity	6 ore	Inquadramento della professione <ul style="list-style-type: none"> <li>▪ Orientamento al ruolo</li> <li>▪ Elementi di diritto del lavoro, contrattualistica, regimi fiscali e responsabilità civile</li> </ul>
	22 ore	Inquadramento dei sistemi informatici <ul style="list-style-type: none"> <li>▪ Fondamenti di informatica: architetture, sistemi operativi; principi di programmazione, relazioni fra algoritmi e dati, tipologie di linguaggi</li> <li>▪ Principi di tutela dei dati personali (2h)</li> </ul>
	15 ore	Basi di cybersecurity <ul style="list-style-type: none"> <li>▪ Principi e tecniche di cybersecurity, migrazione e protezione dei dati</li> <li>▪ Introduzione alle tecniche di test per la sicurezza informatica</li> <li>▪ Cenni sulla crittografia simmetrica e asimmetrica</li> </ul>
MODULO 1. Scelta delle modalità di sviluppo del software	37 ore	Definire le modalità di sviluppo del Software <ul style="list-style-type: none"> <li>▪ Ciclo dello sviluppo del software. Principi e tecniche di project management applicati allo sviluppo di software</li> <li>▪ Metodi e strumenti per la realizzazione del software</li> <li>▪ Le fasi del ciclo di vita del software:               <ul style="list-style-type: none"> <li>○ Analisi: Individuare requisiti e funzionalità del sistema software</li> <li>○ Progettazione: architettura del sistema e funzionamento dei moduli software che costituiscono il sistema</li> <li>○ Programmazione: fase in cui si codifica la soluzione progettata nella fase precedente</li> <li>○ Test: in questa fase si progettano e si eseguono i vari casi di test partendo dai criteri di validazione espressi nell'analisi dei requisiti fino alla loro realizzazione nel codice</li> <li>○ Deployment: in questa fase si rende il software disponibile all'utente e quindi eseguibile</li> <li>○ Manutenzione: questa è la fase in cui si adegua o si corregge il software</li> <li>○ Attraverso le fasi di sviluppo del software si acquisirà padronanza di linguaggi di modellazione quali UML e strumenti o tool di supporto</li> </ul> </li> <li>▪ Tipologie di strumenti di supporto all'organizzazione e alla gestione del processo di sviluppo del software (IDE, repository, librerie condivise, strumenti di debugging, versioning, testing, workflow, supporto delle community online). Principi di pianificazione e disegno di testing e criteri di validazione e riuso del software</li> <li>▪ Principi, metodi e strumenti di gestione della documentazione dello sviluppo del software</li> </ul> <p>Si acquisiranno i principi base per la progettazione strutturata e quella object oriented. Cenni alle varie forme di processo di sviluppo:</p> <ul style="list-style-type: none"> <li>- iterativo e incrementale</li> <li>- a cascata</li> </ul> <p>Nella fase di progettazione si analizzeranno alcuni dei pattern di disegno più usati: Singleton, Factory, Builder, Adapter, State, che verranno poi ripresi e implementati nel modulo successivo.</p>



		<p>Possibili strumenti:</p> <p>IDE (Integrated Development Environment):</p> <ul style="list-style-type: none"><li>- NetBeans</li><li>- Eclipse</li><li>- Visual Studio Code</li></ul> <p>Programmazione condivisa e salvataggio su cloud:</p> <ul style="list-style-type: none"><li>- GIT</li><li>- Bitbucket</li><li>- Github</li></ul>
<p>MODULO 2. Sviluppo delle componenti software e delle interfacce</p>	<p>112 ore</p>	<p>Svolgere attività di coding - Linguaggi ad oggetti e procedurali</p> <ul style="list-style-type: none"><li>▪ Framework di programmazione</li><li>▪ Linguaggi di programmazione ad oggetti e procedurali</li><li>▪ Strumenti a supporto del coding</li></ul> <p>FONDAMENTI DI PROGRAMMAZIONE</p> <p>Obiettivi: Acquisire competenze nella scrittura di algoritmi a prescindere dal linguaggio di programmazione; Acquisire competenze di programmazione strutturata in termini di tecniche e principi.</p> <p>Contenuti: Algoritmo e dato; concetto di variabile; strutture base: selezione, iterazione e sequenza</p> <p>JAVA BASE</p> <p>Obiettivi: Imparare a realizzare programmi a partire da input provenienti dalla progettazione.</p> <p>Contenuti: Struttura di un programma java; utilizzo delle variabili; operatori ed espressioni; selezione, iterazione e sequenza in java; codifica degli algoritmi teorizzati; classi e oggetti in java; attributi e metodi; parametri e valori restituiti da metodi; costruttori; ereditarietà, information hiding e polimorfismo nel linguaggio java; interface e classi abstract; classi anonime e interne; classi wrapper; eccezioni e loro gestione; differenze tra reference e oggetti. Elementi di debug.</p> <p>JAVA AVANZATO</p> <p>Obiettivi: Realizzare programmi con strumenti avanzati a partire da input di progettazione</p> <p>Contenuti: Tipi enumerati; Collection: Set, List, e Map e relative classi concrete; Ordinamenti di oggetti; Interface comparable; Interface comparator; Generics; Nested class; Espressioni Lambda, Reference a metodi, Gestione File, Interazione con DBMS attraverso JDBC. Elementi di debug.</p> <p>JAVA WEB</p> <p>Obiettivi: Realizzare applicazione web usando la tecnologia JEE</p>

		<p>Contenuti: Modello client-server; pattern MVC; servlet; JSP – Java Server Page; EL: expression language; JSTL: Standard Tag Library; definizione di un connection pool su AS GlassFish e utilizzo di DataSource.</p> <p>Possibili strumenti: Apache Web Server, TOMCAT Application Server, ORACLE e MYSQL DBMS, GlassFish. Uso di Docker Containers per il testing e il deploying di applicazioni client e server side.</p>
	15 ore	<p>Svolgere attività di coding – Linguaggio di scripting e markup</p> <ul style="list-style-type: none"> <li>▪ Linguaggi di scripting. Metalinguaggi di markup <ul style="list-style-type: none"> <li>○ Linguaggio HTML. Linguaggio XML e cenni su architetture SOAP e scrittura di WSDL. Linguaggio JSON e struttura dei web service REST.</li> </ul> </li> </ul>
	22 ore	<p>Applicare principi di business logic e responsive design</p> <ul style="list-style-type: none"> <li>▪ Principi di business logic e di responsive design <ul style="list-style-type: none"> <li>○ Programmazione MVC (model, view, controller) e strumenti implementativi in Java (JSP, servlet, DAO o hibernate per la parte model)</li> <li>○ Strumenti di responsive design (mobile, pc, tablet, etc.) per la realizzazione di interfacce web e il testing multiplatforma (RWD)</li> </ul> </li> </ul>
MODULO 3. Modellazione, interrogazione e gestione dei dati	22 ore	<p>Applicare principi di universal design</p> <ul style="list-style-type: none"> <li>▪ Principi di universal design (interfacce standardizzate, adattabilità dei prodotti), ergonomia ed accessibilità, User Experience <ul style="list-style-type: none"> <li>○ Sviluppo di interfacce basate su riconoscimento vocale orientato all'accessibilità dei prodotti per garantire la piena fruibilità dell'esperienza di utilizzo da parte di tutte le tipologie di utilizzatori. (Google text-to-speech, SynthesizeSpeech)</li> </ul> </li> </ul>
	112 ore	<p>Modellizzare, interrogare e gestire i dati</p> <ul style="list-style-type: none"> <li>▪ Tipologie di DBMS - database management system: relazionali. Linguaggi strutturati e specifici di interrogazione. Procedure di storage, in DBMS relazionali. Metodi e tecniche di standardizzazione e diffusione dei dati</li> </ul> <p>PROGETTAZIONE BASE DATI</p> <ul style="list-style-type: none"> <li>○ Obiettivi: Apprendere i principi di progettazione di un Database; capire come normalizzare e ottimizzare un DB (Database).</li> <li>○ Contenuti: Definizione requisiti: informativi, sui processi, sui vincoli di integrità. Diagrammi entità-relazioni. SQL base. Progettazione concettuale: schema concettuale; Progettazione logica: schema logico; Progettazione fisica: schema fisico</li> </ul> <p>JAVA SPRING E ANGULAR</p> <ul style="list-style-type: none"> <li>○ Obiettivi: Imparare a realizzare applicazioni web usando framework Spring e Angular</li> <li>○ Contenuti – Spring e Angular in Java: front-end e back-end, funzioni e loro interazioni. Creare servizi REST lato Spring. Creare interfacce grafiche lato Angular. Richiamare servizi REST da Angular</li> </ul>
MODULO 4.	22 ore	<p>Testare il prodotto sviluppato</p> <ul style="list-style-type: none"> <li>▪ Metodi e tecniche di disegno di test di prodotto</li> </ul>



Testing e Deployment del software sviluppato		<ul style="list-style-type: none"><li>▪ Realizzare test di prodotto preconsegna, sulla scorta delle specifiche tecniche, in termini di funzionalità e interoperabilità<ul style="list-style-type: none"><li>○ Obiettivi: Presentare le strategie di pianificazione ed esecuzione dei test; Introdurre alle tecniche di test per la sicurezza informatica</li><li>○ Contenuti: Piano dei test, Tecniche statiche, Tecniche dinamiche, Inspect code, Livelli di test: unitario, di modulo, di sottosistema, di sistema, di integrazione, alfa-test, beta-test, Stress test, Test di regressione, Errori indotti dalla manutenzione, Strategie di test, Ripetibilità dei test</li></ul></li></ul>
	37 ore	<ul style="list-style-type: none"><li>Definire e condurre il Deployment</li><li>▪ Metodi e tecniche di deployment, definizione del workflow</li><li>▪ Principi e tecniche di configurazione di componenti hardware e software</li><li>▪ Strumenti per la migrazione dei dati</li><li>▪ Principi e tecniche di produzione e organizzazione della documentazione</li><li>▪ Metodi e tecniche di formazione addestrativa degli utenti<ul style="list-style-type: none"><li>○ Obiettivi: Pianificare la migrazione e il backup dei dati, garantendo integrità e riservatezza Redigere la documentazione, registrando tutte le informazioni rilevanti</li><li>Formare o supportare la formazione degli utenti sull'uso della soluzione rilasciata, illustrando le situazioni tipo e i metodi per risolvere gli errori più comuni</li><li>○ Contenuti: Docker, installazione, creazione e configurazione open source di ambienti virtuali personalizzati (rispetto all'applicazione oggetto del test e rispetto agli ambienti di deploying previsti per l'applicazione stessa) per il testing e il deployment di applicazioni su diverse piattaforme.</li></ul></li></ul>
MODULO SICUREZZA Sicurezza sul Lavoro	8 ore	<ul style="list-style-type: none"><li>▪ Operare in sicurezza nel luogo di lavoro<ul style="list-style-type: none"><li>○ Legislazione sulla salute e sicurezza sui luoghi di lavoro e applicazione delle norme di sicurezza.</li><li>○ Gli obblighi del datore di lavoro e del lavoratore.</li><li>○ Dispositivi di protezione individuali (DPI)</li></ul></li></ul>
Tirocinio Curriculare	140 ore	<ul style="list-style-type: none"><li>▪ Attività Formativa in Azienda. Obiettivi formativi:</li><li>▪ Realizzare programmi con strumenti avanzati a partire da input di progettazione formalizzato con un linguaggio, come UML</li><li>▪ Definire i requisiti utente di progetto</li><li>▪ Definire output di progettazione di SW e di dati</li><li>▪ Realizzare applicazioni web usando anche framework quali Spring e Angular</li></ul>
Totale corso	570 ore	Gli allievi sono ammessi all'esame finale per l'acquisizione della Qualifica Professionale di "Programmatore Informatico" (K1.12 del Repertorio Regionale del Lazio) a condizione di avere frequentato almeno l'80% delle ore complessive del percorso formativo
MODULO Integrativo Coaching & Job Placement	30 ore	<ul style="list-style-type: none"><li>▪ 8 ore Orientamento Specialistico</li><li>▪ 22 ore Accompagnamento al Lavoro</li></ul>
Totale percorso	600 ore	