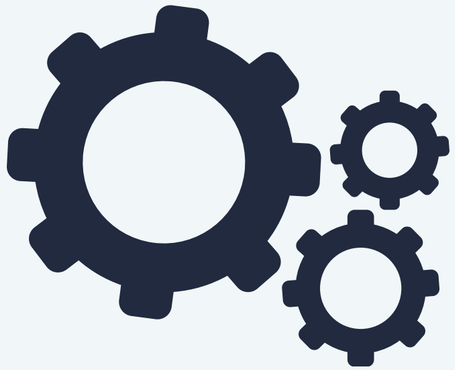
PROGETTO
TYPESCRIPT

DI IOTTI SOFIA

INDICE

01 INTRODUZIONE

IL PROGETTO CONCLUDE IL PERCORSO DI STUDIO SU **TYPESCRIPT** DI **START2IMPACT** UNIVERSITY.
IL CLIENTE IPOTETICO È –**INCLUDO**– AZIENDA CHE SI PROPONE DI CREARE PERCORSI DI FORMAZIONE PROFESSIONALE PER MIGRANTI E PERSONE SVANTAGGIATE PER INCLUDERLE NEL MONDO DEL LAVORO.



IL TEST PREVEDE LA CREAZIONE DI UN SISTEMA DI GESTIONE DELLA SCUOLA DI FORMAZIONE IN GRADO DI ORGANIZZARE LE INTERAZIONI TRA PARTECIPANTI, AZIENDE, CORSI E POSIZIONI DI LAVORO DISPONIBILI.

02 LINK AL PROGETTO E GITHUB

LINK PROGETTO

HTTPS://CODEPEN.IO/SOFIAIOTTI/PEN/POROXWV

LINK REPOSITORY GITHUB

HTTPS://GITHUB.COM/SOFIAIOTTI/TYPESCRIPTPROJECT

03 LINGUAGGIO UTILIZZATO

TYPESCRIPT

04 STRUTTURA DEL PROGETTO

INTERFACCE

LE INTERFACCE DEFINISCONO I TRE ELEMENTI PRINCIPALI DEL SISTEMA PROGETTATO, DECIDENDONE PROPRIETÀ E METODI.

- **IPartecipante** - RAPPRESENTA COLORO CHE DESIDERANO APPLICARSI AI PROGRAMMI DELLA SCUOLA E CHE POTRANNO CONTRARRE I CONTRATTI LAVORATIVI DELLE AZIENDE;
- **ICorso** - RAPPRESENTA I CORSI OFFERTI DAL PROGRAMMA DI INCLUDO;
- **IAzienda** - RAPPRESENTA LE AZIENDE AFFILIATE A INCLUDO CHE OFFRIRANNO IMPIEGHI AI PARTECIPANTI AI CORSI.

```typescript
// Interfacce

interface IPartecipante{
    nome: string;
    cognome: string;
    paeseOrigine: string;
    livelloIstruzione: number;
    competenzeLinguistiche: string;
    ambitoInteresse: string;

    iscrivitiCorso(corso: ICorso): void;
}

interface ICorso{
    titolo: string;
    descrizione: string;
    settore: string;
    durata: number;
    elencoIscritti: IPartecipante[];

    aggiungiPartecipante(partecipante: IPartecipante): void;
    stampaElencoIscritti(): void;
}

interface IAzienda{
    nome: string;
    settore: string;
    descrizione: string;
    posizioniAperte: string[];

    offriPosizione(partecipante: IPartecipante, posizione: string): void;
}
```

CLASSI

LE CLASSI Partecipante, Corso E Azienda IMPLEMENTANO LE CORRISPONDENTI INTERFACCE PRECEDENTEMENTE CREATE GESTENDO LE LOGICHE DI FUNZIONAMENTO DELLA SCUOLA DI FORMAZIONE.

```
// Classi

class Partecipante implements IPartecipante(){
    nome: string;
    cognome: string;
    paeseOrigine: string;
    livelloIstruzione: number;
    competenzeLinguistiche: string;
    ambitoInteresse: string;

    iscrizioni: ICorso[] = [];

    constructor(nome: string, cognome: string, paeseOrigine: string, livelloIstruzione: number,
competenzeLinguistiche: string, ambitoInteresse: string){
        this.nome = nome;
        this.cognome = cognome;
        this.paeseOrigine = paeseOrigine;
        this.livelloIstruzione = livelloIstruzione;
        this.competenzeLinguistiche = competenzeLinguistiche;
        this.ambitoInteresse = ambitoInteresse;
    }

    iscrivitiCorso(corso: ICorso): void{
        corso.aggiungiPartecipante(this);
        this.iscrizioni.push(corso);
        console.log(`${this.nome} ${this.cognome} è ora iscritto al corso: "${corso.titolo}".`);
    }
}
```

```
class Corso implements ICorso {
    titolo: string;
    descrizione: string;
    settore: string;
    durata: number;

    elencoIscritti: IPartecipante[] = [];

    constructor(titolo: string, descrizione: string, settore: string, durata: number) {
        this.titolo = titolo;
        this.descrizione = descrizione;
        this.settore = settore;
        this.durata = durata;
    }

    aggiungiPartecipante(partecipante: IPartecipante): void {
        this.elencoIscritti.push(partecipante);
             console.log(`${partecipante.nome} ${partecipante.cognome} è stato aggiunto al corso:
"${this.titolo}".`);
    }
  stampaElencoIscritti(): void {
        let output = `Elenco degli iscritti al corso "${this.titolo}": `;
        for (let i = 0; i < this.elencoIscritti.length; i++) {
            output += `${this.elencoIscritti[i].nome} ${this.elencoIscritti[i].cognome}`;
            if (i < this.elencoIscritti.length - 1) {
                output += ', ';
            } else {
                output += '.';
            }
        }
       console.log(output);
    }
}

class Azienda implements IAzienda {
    nome: string;
    settore: string;
    descrizione: string;
    posizioniAperte: string[];

    constructor(nome: string, settore: string, descrizione: string, posizioniAperte: string[]) {
        this.nome = nome;
        this.settore = settore;
        this.descrizione = descrizione;
        this.posizioniAperte = posizioniAperte;
    }

    offriPosizione(partecipante: IPartecipante, posizione: string): void {
        console.log(`L'azienda ${this.nome} offre una posizione come ${posizione} a ${partecipante.nome}
${partecipante.cognome}.`);
    }
}
```

ISTANZE E TEST

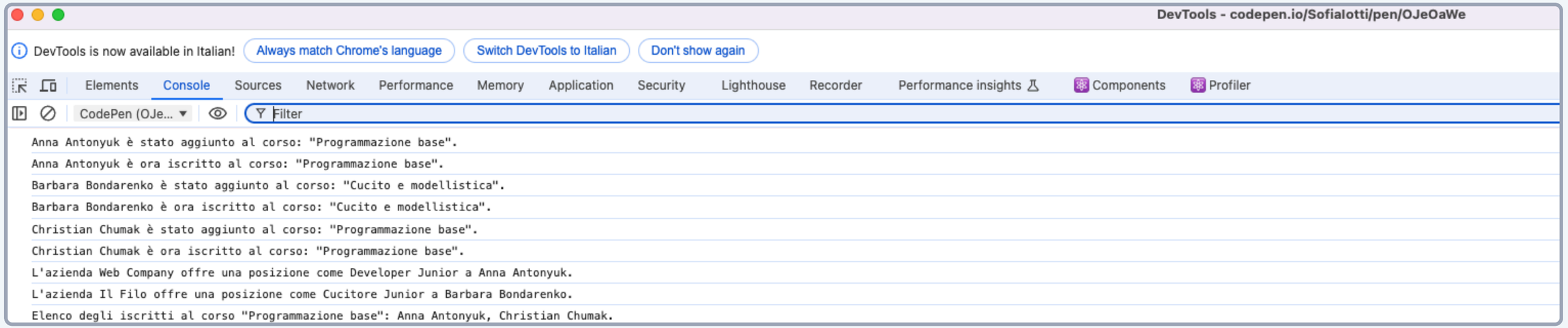
NELLA FASE DI TESTING, HO PROCEDUTO INIZIALMENTE AD ISTANZIARE ALCUNI OGGETTI PER OGNI CLASSE:

- migrante1, migrante2 e migrante 3 per la classe Partecipante;
- corsoInformatica e corsoSartoria per la classe Corso;
- WebCompany e IlFIlo per la classe Azienda.

SUCCESSIVAMENTE HO TESTATO LA LOGICA DI FUNZIONAMENTO DELLE ISCRIZIONI E DELL'OFFERTA DELLE POSIZIONI LAVORATIVE.

```
//Istanze e testing

const migrante1 = new Partecipante('Anna', 'Antonyuk', 'Ucraina', 8, 'Buone', 'Informatica');
const migrante2 = new Partecipante('Barbara', 'Bondarenko', 'Ucraina', 7, 'Ottime',
'Sartoria');
const migrante3 = new Partecipante('Christian', 'Chumak', 'Ucraina', 6, 'Sufficienti',
'Informatica');


const corsoInformatica = new Corso('Programmazione base', 'Studio delle basi della
programmazione', 'Informatica', 100);
const corsoSartoria = new Corso('Cucito e modellistica', 'Studio e confezione base del capo
sartoriale', 'Sartoria', 150);

const WebCompany = new Azienda('Web Company', 'Informatica', 'Sviluppo Software e siti Web.',
['Developer Junior', 'Marketing Specialist']);
const IlFilo = new Azienda('Il Filo', 'Sartoria', 'Produzione artigianale di abiti per donna,
uomo e bambino.', ['Cucitore Junior', 'Addetto allo stiro'])

migrante1.iscrivitiCorso(corsoInformatica);
migrante2.iscrivitiCorso(corsoSartoria);
migrante3.iscrivitiCorso(corsoInformatica);

WebCompany.offriPosizione(migrante1, 'Developer Junior');
IlFilo.offriPosizione(migrante2, 'Cucitore Junior');

corsoInformatica.stampaElencoIscritti();
```

OUTPUT



NELL'IMMAGINE È VISIBILE L'OUTPUT RISULTANTE DALLE CHIAMATE AI METODI DELLE ISTANZE DELLE CLASSI PARTECIPANTE, CORSO E AZIENDA.

05 CONCLUSIONE

GRAZIE PER L'ATTENZIONE

Ogni progetto del master è prova di quanto sia fondamentale mettersi in gioco e sporcarsi le mani per imparare.

Anche questa volta, dare forma allo studio con un esempio pratico mi ha aiutata tantissimo a capire gli argomenti trattati.