

NPO法人ライツオン・チルドレン ジョブリハ「プログラミングの仕事」 参加児童募集のご案内

【プログラムの目的】

・ジョブリハは「ジョブリハーサル」の略で、児童養護施設や里親で暮らす**中高生を対象**に、特定の業種を舞台にして**仕事体験**を提供するプログラムです。

・進路を選択する前に、社会人や仕事と触れ合っていくことで、よりスムーズな自立を促します。また、1年かけて仕事に関わることで、仕事の楽しさや苦勞について学び、「働く」ことについての理解やイメージを深めることを目的としています。

【「プログラミングの仕事」の内容】

「プログラミングの仕事」では、**簡単なゲームを作成しながら楽しくプログラミング**について学びます。

【参加費用】

ジョブリハで使用する**パソコンは参加者にプレゼント**します。参加費は**無料**です。（※インターネット通信料は各自ご負担頂きます）

こんな子どもが対象です！

- ・将来どんな仕事をしたいか悩んでいる。
 - ・色々経験してみたい！
 - ・パソコンを使った仕事に興味がある。
 - ・ゲームをするのが好き！ゲームの開発者になりたい。
 - ・進学先の専攻分野を考え中。
- まずは気軽に説明会に参加してみてください♪

申し込み～開催スケジュール予定

施設職員・里親向け説明会（3/11・15・19）



子ども向け説明会（4/5・10）



申し込み期間 4/20～28（定員9名、先着順）



5月初旬 参加児童決定、パソコン発送



前半

第1回（5/29） 第2回（6/4） 第3回（7/3）
第4回（8/4） 13:00～16:00（3時間）



後半

第5回（9/4） 第6回（10/1） 第7回（11/6）
第8回（12/3） 第9回（1/8） 第10回（2/4）
11:00～13:00／休憩／14:00～16:00（4時間）

・講習会はオンライン形式（ZOOM）での開催となりますが、第4回と第10回は渋谷にて対面形式で発表会を開催予定です。（感染状況によりオンライン形式への変更あり）

・プログラムは1年間全10回で、前半と後半がありますが、前半のみの参加も可能です。

ジョブリハ「プログラミングの仕事」 施設職員・里親向け説明会のご案内

【開催日時】

- ① 3月11日 (金) 11:00～12:00
- ② 3月15日 (火) 14:00～15:00
- ③ 3月19日 (土) 13:00～14:00

【開催形式】

オンライン形式（ZOOMミーティング）にて実施。

【対象】

里親様、施設職員様（職種に限らずご参加頂けます）また、複数人での参加も可能です。

【参加方法】 Googleフォームにてお申し込みをお願いします。（締め切り

〇月〇日）※**フォームリンク**

→申し込みフォームを受付けましたら、説明会当日のZOOMのリンクをメールにて別途ご案内致します。

【注意事項】

ジョブリハ「プログラミングの仕事」に参加を希望される場合は、上記説明会には必ずご参加頂けますようお願い致します。**説明会にご参加頂けない場合には、プログラムへの参加は出来ません**のでご了承下さい。

【問い合わせ】

特定非営利活動法人 ライツオン・チルドレン

ジョブリハ プログラミングの仕事 担当

メール：jr.p@lightson-children.com

オンラインで参加するってどんな感じ？

- ・ 基本的にはご自宅（施設）からの参加を想定しています。
- ・ 講習会ではZOOMミーティングを使用し、講師と子ども、お互いの顔が映る状態で画面を共有しながら、作業を進めことが出来るので安心です。
※カメラに顔を映して参加することが出来ないお子様のご参加はご遠慮頂いております。ご了承ください。
- ・ 集団講義形式ではないので、子ども一人ひとりのペースや能力に合わせて、作業を進めることが出来ます。
- ・ 少人数（講師1人：児童2～3人）グループで作業を行うことで、質問しやすい雰囲気作りを工夫しています。

Zoom

その操作は・・・

全画面表示の開始

教師

この画面の部分が分かりません。

参加児童②

参加児童①

子どもが作業している画面が講師にも共有されるので、理解しやすい。

```
public class GoalChecker : MonoBehaviour
{
    // Start is called before the first frame update
    public GameObject unitChan;
    public AudioSource audioSrc;
    public GameObject goalPost;
    public GameObject redBall;

    void Start()
    {
        redBall.GetComponent<SphereCollider>().enabled = false;
    }

    // Update is called once per frame
    void Update()
    {
        private void OnTriggerEnter(Collider other)
        {
            if (other.gameObject.GetComponent<CapsuleCollider>().isReady == true)
            {
                unitChan.transform.LookAt(goalPost.transform);
                unitChan.GetComponent<Animator>().SetTrigger("Goal");
                audioSrc.Play();
                goalPost.Force();
            }
        }

        redBall.GetComponent<SphereCollider>().enabled = true;
    }

    public void RetireStage()
    {
        SceneManager.LoadScene(SceneManager.GetActiveScene().name);
    }
}
```

ミュート解除 ビデオの開始 セキュリティ 参加者の管理 チャット 画面を共有 レコーディング 反応

ミーティングの終了