

ביטויים רגולריים:

(1)

`\b(?:Jan|Feb|Mar|Apr|May|Jun|Jul|Aug|Sep|Oct|Nov|Dec)[a-z]*\s\d{1,2},\s\d{4}\b`

הסבר:

- מתחיל בשם חודש מקוצר (Jan-Dec)
- אחריו יום (מספר),
- ואז פסיק ורווח,
- ואז שנה של 4 ספרות.

(2)

`\b(?:25[0-5]|2[0-4]\d|1\d\d|[1-9]?d)\.){3}(?:25[0-5]|2[0-4]\d|1\d\d|[1-9]?d)\b`

הסבר:

- 4 קבוצות של מספרים בין 0 ל-255
- כל קבוצה מופרדת בנקודה.

(3)

`<h1[^>]*slotTitle[^>]*>\s*<span[^>]*>\s*(.*?)\s*`

הסבר:

- מאתר תגית `<h1>` עם המילה `slotTitle`
- בפנים יש `` עם הטקסט של הכותרת.

(4)

`href=([>]*download[>]*)`

הסבר:

- מחפש `href=` עם מחרוזת שיש בה את המילה `download`
- מוצא את הלינק של ההורדה.

(5)

`<a[^>]*href=[>]*jmeter[^>]*\.html[^>]*>(.*?)`

הסבר:

- מחפש תגית `<a>` עם קישור שמכיל `jmeter.html`.
- שולף את הטקסט שבתוך התגית.

(6

<span[^>]*>[^<]*\s*([\u0590-\u05FF\s]+)

הסבר:

- מחפש תגית
- ואז תווים בעברית (תחום יוניקוד של עברית).

(7

>(NFT jobs)<

הסבר:

- בודק אם יש טקסט שמכיל בדיוק את "NFT jobs" בין תגיות HTML.

(8

>(*[>^]software[^<]*non[^<]*functional[^<]*testing*[>^])<

הסבר:

- מחפש טקסט גלוי שכולל את כל המילים האלה לפי הסדר
- גם אם יש תווים ביניהם (כמו רווחים או נקודות).